Modern Computer Algebra

Solutions to selected exercises

14 September 2003

JOACHIM VON ZUR GATHEN
and
JÜRGEN GERHARD

Universität Paderborn

CAMBRIDGE
UNIVERSITY PRESS

Contents

Chapter 2

2.1 Since $a_{l-1} \neq 0$, we have $r^{l-1} \leq a = \sum_{0\leq i<l} a_i r^i \leq \sum_{0\leq i<l}(r-1)r^i = r^l - 1$, and taking logarithms, we obtain $l - 1 \leq \log_r a < l$. Since l is an integer, this is equivalent to $l - 1 = \lfloor \log_r a \rfloor$.

2.3 (i) The single precision subtraction instruction takes two single precision integers a and b and the contents of the carry flag γ as input, and outputs c and sets the carry flag γ^* such that $a - b - \gamma = -\gamma^* \cdot 2^{64} + c$.

(ii) ALGORITHM 2.6 Subtraction of multiprecision integers.
Input: Two multiprecision integers $a = (-1)^s \sum_{0\leq i\leq n} a_i 2^{64i}$, $b = (-1)^s \sum_{0\leq i\leq n} b_i 2^{64i}$, not necessarily in standard representation, with $|a| > |b|$ and $s \in \{0,1\}$.
Output: $c = (-1)^s \sum_{0\leq i\leq n} c_i 2^{64i}$ such that $c = a - b$.

1. $\gamma_0 \longleftarrow 0$
2. **for** $i = 0, \ldots, n$ **do**
   $c_i \longleftarrow a_i - b_i - \gamma_i, \quad \gamma_{i+1} \longleftarrow 0$
   if $c_i < 0$ **then** $c_i \longleftarrow c_i + 2^{64}, \quad \gamma_{i+1} \longleftarrow 1$
3. **return** $(-1)^s \sum_{0\leq i\leq n} c_i 2^{64i}$

(iii) With $a = (-1)^s \sum_{0\leq i\leq n} a_i 2^{64i}$ and $b = (-1)^s \sum_{0\leq i\leq n} b_i 2^{64i}$, we have $|a| > |b|$ if and only if $a_n = b_n, a_{n-1} = b_{n-1}, \ldots, a_{n-i+1} = b_{n-i+1}, a_{n-i} > b_{n-i}$ for some $i \in \{0, \ldots, n\}$. Each comparison is essentially a single precision subtraction.

2.5 ALGORITHM 2.7 Multiplication by a single precision integer.
Input: $a, b \in \mathbb{Z}$ such that $b = (-1)^s \sum_{0\leq i\leq m} b_i 2^{64i}$, with $s \in \{0,1\}$ and $a, b_0, \ldots, b_m \in \{0, \ldots, 2^{64} - 1\}$.
Output: The multiprecision integer $ab \in \mathbb{Z}$.

1. compute $c_0, w_1 \in \{0, \ldots, 2^{64} - 1\}$ such that $a \cdot b_0 = w_1 2^{64} + c_0$
   $\gamma_1 \longleftarrow 0$
2. **for** $i = 1, \ldots, m$ **do**
3. compute $u_i, w_{i+1} \in \{0, \ldots, 2^{64} - 1\}$ such that $a \cdot b_i = w_{i+1} 2^{64} + u_i$
4. compute $c_i \in \{0, \ldots, 2^{64} - 1\}$ and $\gamma_{i+1} \in \{0,1\}$ with $\gamma_{i+1} 2^{64} + c_i = u_i + w_i + \gamma_i$
5. $c_{m+1} \longleftarrow w_{m+1} + \gamma_{m+1}$
   return $(-1)^s \sum_{0\leq i\leq m+1} c_i 2^{64i}$

2.7 We have $\lambda(a) - \lambda(b) \leq \lambda(q) \leq \lambda(a) - \lambda(b) + 1$. The bounds are achieved when $a = 2^{64(m-1)}$, $b = 2^{64n} - 1$ and $a = 2^{64m} - 1$, $b = 2^{64(n-1)}$, respectively.

2.8 Suppose to the contrary that $x^2 = q\cdot(2x+1)+r$, with $q,r \in \mathbb{Z}[x]$ and $\deg r < 1$. Comparing leading coefficients, we find that $1 = \mathrm{lc}(q)\cdot 2$, which is impossible since $\mathrm{lc}(q) \in \mathbb{Z}$.

2.9 We replace step 3 of Algorithm 2.5 by

3. **if** $\deg r = m+i$ **then**
   if $b_m \mid \mathrm{lc}(r)$ **then** $q_i \longleftarrow \mathrm{lc}(r)/b_m, \quad r \longleftarrow r - q_i x^i b$
   else return "FAIL"
   else $q_i \longleftarrow 0$

This proves existence of q and r in case that the inner condition is always true. Uniqueness follows from the uniqueness of q and r over K. Conversely, let $i \le n-m$ be the largest index such that $\deg r = m+i$ and $b_m \nmid \mathrm{lc}(r)$, and suppose that there exist $q^*, r^* \in R[x]$ such that $a = q^* b + r^*$ and $\deg r^* < \deg b$. By the invariant $a = (\sum_{i<j\le n-m} q_j x^j) b + r$, which holds at the beginning of step 3, we find that $r = q^{**} b + r^*$, where $q^{**} = (q^* - \sum_{i<j\le n-m} q_j x^j)$. Comparing leading coefficients leads to the contradiction $\mathrm{lc}(r) = \mathrm{lc}(q^{**}) b_m$.

Chapter 3

3.3 We prove all statements for an integral domain R in which any two elements have a gcd and a lcm, with $|\cdot|$ replaced by $\mathrm{normal}(\cdot)$.

(i) By definition of the gcd, we have that $\gcd(a,b) \mid a$. If $a \mid b$, then a is a common divisor of a and b, and hence $a \mid \gcd(a,b)$. Thus $\gcd(a,b) = \mathrm{normal}(a)$ since both elements are normalized. Conversely, if $\mathrm{normal}(a) = \gcd(a,b)$, then $a = \mathrm{lu}(a)\gcd(a,b)$, and a divides b.

(ii) follows from (i) with $b = a$, $b = 0$, and $a = 1$, respectively. (This part of the exercise is only present in the 2003 edition.)

(iii) is immediate from the definition.

(iv) Every common divisor of a,b,c divides both sides of (iii), and both sides of (iii) are common divisors of a,b,c. The claim follows from both sides being normalized.

(v) The claim is clear if $c = 0$, and we may assume that $c \neq 0$. If $d \in R$ divides a and b, then $\mathrm{normal}(c)d$ divides ca and cb, and $\gcd(ca,cb)$. In particular, this holds for $d = \gcd(a,b)$. Conversely, c is a common divisor of ca and cb, and hence c divides $d = \gcd(ca,cb)$. Let $ca = da^*$, $cb = db^*$, and $d = cd^*$ with $a^*, b^*, d^* \in R$. Then $ca = cd^*a^*$ and $cb = cd^*b^*$, and since R is an integral domain, this implies that $a = d^*a^*$ and $b = d^*b^*$, so that d^* is a common divisor of a and b. Thus $d^* \mid \gcd(a,b)$ and $d = cd^* \mid \mathrm{normal}(c)\gcd(a,b)$. The claim follows since both sides are normalized.

(vi) If $\mathrm{normal}(a) = \mathrm{normal}(b)$, then a and b have the same divisors. Thus for $d \in R$, we have

$$d \mid \gcd(a,c) \iff d \mid a \text{ and } d \mid c \iff d \mid b \text{ and } d \mid c \iff d \mid \gcd(b,c).$$

All claims (and their proofs) remain valid when the gcd is replaced by the lcm and all divisibility statements are "reversed".

3.4 $\gcd(a,bc) = \gcd(\gcd(a,ab),bc) = \gcd(a,\gcd(ab,bc)) = \gcd(a,\gcd(a,c)b) = \gcd(a,b) = 1$.

3.5 (ii) Let $a,b \in R$ with $b \neq 0$ and $\delta \in D$ be such that $d(b) = \delta(b)$. Since δ is a Euclidean function, there exist $q,r \in R$ such that $a = qb+r$ and $\delta(r) < \delta(b)$. By the definition of d, we have $d(r) \leq \delta(r) < \delta(b) = d(b)$.

(iii) We define δ^* by $\delta^*(b) = \delta(ab)$ and $\delta^*(r) = \delta(r)$ if $r \neq b$. Then $\delta^*(r) \leq \delta(r)$ for all $r \in R$, and we have to show that δ^* is a Euclidean function. Let $f,g \in R$ such that $g \neq 0$. If $g \neq b$, then there exist $q,r \in R$ such that $f = qg+r$ and $\delta^*(r) \leq \delta(r) < \delta(g) = \delta^*(g)$. If $g = b$, then there exist $q,r \in R$ such that $f = qab+r$ and $\delta^*(r) \leq \delta(r) < \delta(ab) = \delta^*(b)$.

(iv) Let $a,b \in R \setminus \{0\}$. Dividing a by itself with remainder, we find $q,r \in R$ such that $a = qa+r$ and $d(r) < d(a)$. If $r \neq 0$, then $q \neq 1$ and $d(r) = d((1-q)a) \geq d(a)$, and this contradiction shows that $r = 0$ and $d(0) < d(a)$.

If $a \in R^\times$, then $d(ab) \geq d(b)$ and $d(b) = d(a^{-1}(ab)) \geq d(ab)$, whence $d(ab) = d(b)$. Conversely, suppose that $d(ab) = d(b)$. Then there exist $q,r \in R$ such that $b = q(ab)+r$ and $d(r) < d(b)$. But $d(r) = d((1-qa)b) \geq d(b)$ if $1 \neq qa$, and hence $r = 0$ and $1 = qa$.

(v) Let δ be a Euclidean function. Since for all $a \in R$, $a = q \cdot 1 + r$ is satisfied for $q = a$ and $r = 0$, the function δ^* defined by $\delta^*(0) = -\infty$, $\delta^*(1) = 0$, and $\delta^*(b) = \delta(b)$ for $b \neq 0,1$ is also a Euclidean function, whence $d(0) = -\infty$ and $d(1) = 0$. By (iv), we have $d(a) = d(a \cdot 1) = d(1) = 0$ if and only if $a \in R^\times$.

(vi) By the minimality of d, we have $d(b) \leq \deg b$ for all nonzero $b \in F[x]$. Suppose that $n = d(b) < \deg b$ for some nonconstant $b \in F[x]$, and that n is minimal with this property. Then there exist $q,r \in F[x]$ such that $x^n = qb+r$ and $\deg r = d(r) < n < \deg b$. Comparing degrees on both sides, we see that $q = 0$ and $r = x^n$. But then $d(r) < n = \deg r$, contradicting the minimality of n, and we conclude that $d = \deg$.

In the integer case, for all $a,b \in \mathbb{Z}$ such that $b \neq 0$ we may find $q,r \in \mathbb{Z}$ such that $a = qb+r$ and $|a| \leq |b|/2$. This proves that $\delta(b) = \lfloor \log_2 |b| \rfloor$ is a Euclidean function. The proof that $d = \delta$ is analogous to the polynomial case, with x replaced by 2.

3.6 (i) Let $p_1,\ldots,p_r$ be the normal forms of all pairwise non-associate prime divisors of ab. Then $\text{normal}(a) = \prod_{1\leq i\leq r} p_i^{e_i}$ and $\text{normal}(b) = \prod_{1\leq i\leq r} p_i^{f_i}$ for some integers $e_1,\ldots,e_r, f_1,\ldots,f_r \in \mathbb{N}$, and $\prod_{1\leq i\leq r} p_i^{\min\{e_i,f_i\}}$ is the gcd and $\prod_{1\leq i\leq r} p_i^{\max\{e_i,f_i\}}$ is the lcm of a and b.

(ii) This follows from (i) and the fact that $\max(e_i,f_i) + \min(e_i,f_i) = e_i + f_i$ for all i.

(iii) We show $\gcd(a_1 \cdots a_{n-1}, a_n) = 1$ and the claim simultaneously by induction on n. The case $n = 2$ is (ii). For $n > 2$, Exercise 3.4 with $a = a_n$, $b = a_1 \cdots a_{n-2}$,

and $c = a_{n-1}$, and the induction hypothesis imply that $\gcd(a_1 \cdots a_{n-2} \cdot a_{n-1}, a_n) = 1$. Thus

$$\begin{aligned}\operatorname{lcm}(a_1, \ldots, a_{n-1}, a_n) &= \operatorname{lcm}(\operatorname{lcm}(a_1, \ldots, a_{n-1}), a_n) \\ &= \operatorname{lcm}(\operatorname{normal}(a_1 \cdots a_{n-1}), a_n) \\ &= \operatorname{normal}(a_1 \cdots a_{n-1}) \operatorname{normal}(a_n) = \operatorname{normal}(a_1 \cdots a_n),\end{aligned}$$

by the induction hypothesis and (ii).

(iv) No; a counterexample is $a_1 = 6$, $a_2 = 10$, $a_3 = 15$ in $R = \mathbb{Z}$.

3.7 (i) follows from the first property and the surjectivity of d, using Exercise 3.5 (iv).

(ii) The first two properties imply that F is closed under $+$ and $\cdot$. By (i), we have $0 \in F$, and $1 \in F$ follows from $d(1) = d(1 \cdot 1) = d(1) + d(1)$. Finally, Exercise 3.5 (iv) implies that $R^\times = F \setminus \{0\}$.

(iii) We prove existence by induction on $n = d(a)$. This is clear for $n = 0$. If $n > 0$, then we divide a by x^n with remainder and obtain $a_n, r \in R$ with $a = a_n x^n + r$ and $d(r) < n$. Then clearly $a_n \neq 0$, and the first two properties imply that $d(a_n) = 0$. Inductively, we find that $r = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ and obtain a representation of a as required. For the uniqueness, it is sufficient to prove that $a = 0$ has no representation of the required form with $n \geq 0$. This follows from the first two properties.

3.10 No.

3.11 (i) 1; (ii) 17; (iii) 13; (iv) 7.

3.14 (i) 1 for $p = 2$, $x + 2$ for $p = 3$.

(ii) $x^2 + 1$ for $p = 2$, $x + 1$ for $p = 3$.

(iii) $x + 4$.

(iv) $x + 1$ for $p = 3$, $x^2 + 3x + 2$ for $p = 5$.

3.15 It is clear that all quotients are positive. We only prove the claim about the s_i, using induction on i. For $i = 1$, we have $s_{2i} = s_2 = s_0 - q_1 s_1 = 1 > 0$ and $s_{2i+1} = s_3 = s_1 - q_2 s_2 = -q_2 < 0$. For $i > 1$, we use the induction hypothesis to conclude that $s_{2i} = s_{2i-2} - q_{2i-1} s_{2i-1}$ is positive and $s_{2i+1} = s_{2i-1} - q_{2i} s_{2i}$ is negative. An alternative proof is by using Exercise 3.20 (iv).

Similarly, we show that $1 \leq |s_i| < |s_{i+1}|$ for $i \geq 3$. If $i = 2$, then

$$|s_3| = |s_1 - q_2 s_2| = q_2 |s_2| \geq |s_2| = 1.$$

If $i \geq 3$, then

$$|s_{i+1}| = |s_{i-1} - q_i s_i| = |s_{i-1}| + q_i |s_i| > q_i |s_i| \geq |s_i| \geq 1,$$

by what we have shown above and since $|s_{i-1}| > 0$, by induction.

3.17 We have $s\cdot 2+t\cdot x=\gcd(2,x)=1$, and substituting 0 for x yields the contradiction $s(0)\cdot 2=1$.

3.18 Let (f,g) and (f^*,g^*) in $(R\setminus\{0\})^2$ be two pairs with the same Euclidean representation $(\rho_0,\ldots,\rho_\ell,r_\ell,q_1,\ldots,q_\ell)$. Then inductively $r_{\ell-1}=q_\ell r_\ell$, $r_{\ell-2}=q_{\ell-1}r_{\ell-1}+\rho_\ell r_\ell$, …, $r_1=q_2r_2+\rho_3r_3$, and $r_0=q_1r_1+\rho_2r_2$ are the remainders in the Euclidean Algorithm for both pairs. Thus $(f,g)=(\rho_0r_0,\rho_1r_1)=(f^*,g^*)$, which proves injectivity. For the surjectivity, let $\ell\in\mathbb{N}_{>0}$, $\rho_0,\ldots,\rho_\ell\in F\setminus\{0\}$, $r_\ell,q_1\in R$ nonzero, and $q_2,\ldots,q_\ell\in R$ nonconstant. If we set $r_{\ell+1}=0$ and inductively define $r_{\ell-1},r_{\ell-2},\ldots,r_1,r_0\in R$ as above, then $\deg r_{i+1}<\deg r_i$ and $r_{i-1}=q_ir_i+\rho_{i+1}r_{i+1}$ is a division with remainder for $1\le i\le\ell$. We have $\operatorname{lc}(r_{i-1})=\operatorname{lc}(q_i)\operatorname{lc}(r_i)=\operatorname{lc}(r_i)$ for $1\le i\le\ell$, and inductively, all r_i are monic. Thus $(\rho_0,\ldots,\rho_\ell,r_\ell,q_1,\ldots,q_\ell)$ is the Euclidean representation of (ρ_0r_0,ρ_1r_1). Finally, $\deg\rho_0r_0=\deg q_1+\deg\rho_1r_1\ge\deg\rho_1r_1$, and ρ_1r_1 is nonzero since ρ_1 and r_ℓ are.

3.19 (ii) The units are $\pm1,\pm i$.

(iv) The four gcds and their representations as linear combinations are $1+i=-i\cdot6+(1+2i)(3+i)$, $1-i=-1\cdot6+(2-i)(3+i)$, $-1-i=i\cdot6+(-1-2i)(3+i)$, $-1+i=1\cdot6+(-2+i)(3+i)$.

(v) One gcd is $89+66i$.

3.23 From Lemma 3.8 (vi) (Lemma 3.8 in the 1999 edition) and the fact that the t_i alternate in sign (Exercise 3.15), we find that

$$\begin{aligned} f&=|r_{i-1}t_i-r_it_{i-1}|=r_{i-1}|t_i|+r_i|t_{i-1}|\ge r_{i-1}|t_i|,\\ g&=|r_{i-1}s_i-r_is_{i-1}|=r_{i-1}|s_i|+r_i|s_{i-1}|\ge r_{i-1}|s_i| \end{aligned}$$

for $1\le i\le\ell+1$.

3.25 (iii) At each recursive call, at least one of $\log_2 a$ and $\log_2 b$ is diminished by at least 1, and hence the recursion depth is at most $\lfloor\log_2 a\rfloor+\lfloor\log b\rfloor\in O(n)$. The cost per step is $O(n)$ word operations.

(iv) ALGORITHM 3.18 Binary Extended Euclidean Algorithm.
Input: $a,b\in\mathbb{N}_{>0}$.
Output: $(s,t)\in\mathbb{Z}^2$ such that $sa+tb=\gcd(a,b)$.

1. **if** $a=b$ **then return** $(1,0)$
2. **if** both a and b are even **then return** $\mathrm{EEA}(a/2,b/2)$
3. **if** exactly one of the two numbers, say a, is even **then**
   $(s^*,t^*)\longleftarrow\mathrm{EEA}(a/2,b)$
   if s^* is even **then return** $(s^*/2,t^*)$ **else return** $((s^*\pm b)/2,t^*\mp a/2)$
4. **if** both a and b are odd and, say, $a>b$, **then**
   $(s^*,t^*)\longleftarrow\mathrm{EEA}((a-b)/2,b)$
   if s^* is even **then return** $(s^*/2,t^*-s^*/2)$
   else return $((s^*\pm b)/2,t^*-(s\pm a)/2)$

3.30 (i) Let $h = \sum_{n\geq 0} G_n x^n \in \mathbb{Q}[[x]]$. We have

$$h = G_0 + G_1 x + \sum_{n\geq 2} (2G_{n-1} + G_{n-2}) x^n = x + 2xh + x^2 h,$$

whence $h = -x/(x^2+2x-1)$. The zeroes of the denominator are $-1 \pm \sqrt{2}$, and the partial fraction expansion is

$$\begin{aligned} h &= \frac{-2-\sqrt{2}}{4} \frac{1}{x+1+\sqrt{2}} + \frac{-2+\sqrt{2}}{4} \frac{1}{x+1-\sqrt{2}} \\ &= \frac{-\sqrt{2}}{4} \frac{1}{1-(1-\sqrt{2})x} + \frac{\sqrt{2}}{4} \frac{1}{1-(1+\sqrt{2})x} \\ &= \frac{\sqrt{2}}{4} \sum_{n\geq 0} \Big((1+\sqrt{2})^n - (1-\sqrt{2})^n \Big). \end{aligned}$$

Since $|1-\sqrt{2}| < 1$, we have $G_n = \frac{\sqrt{2}}{4}((1+\sqrt{2})^n - (1-\sqrt{2})^n) \approx \frac{\sqrt{2}}{4}(1+\sqrt{2})^n$ for large n.

(ii) With $f = G_{n+1}$ and $g = G_n$, the length of the least absolute remainder Euclidean Algorithm for f, g is

$$\ell = n \approx \log_{1+\sqrt{2}} 2\sqrt{2} G_n \in 0.786 \log_2 g + O(1).$$

Thus the worst case length of the standard Euclidean Algorithm is about twice as long as the worst case length of the least absolute remainder Euclidean Algorithm.

3.31 (i) Induction on n.

(ii) Induction on n and (i).

(iii) Induction on n, (i) and (ii).

(iv) Wrong: for example, $F_5 \text{ rem } F_4 = 5 \text{ rem } 3 = 2 \neq 1 = F_1 = F_{5 \text{ rem } 4}$.

(v) Write $n = qk + r$ with $0 \leq r < k$ and use (i) through (iii).

(vi) Follows from (v) by induction along the Euclidean Algorithm.

(vii) Letting $k = n-1$ and $k = n$ in (i), we find that

$$(F_{2n}, F_{2n+1}) = (F_{n-1}F_n + F_n F_{n+1}, F_n^2 + F_{n+1}^2) = (2F_n F_{n+1} - F_n^2, F_n^2 + F_{n+1}^2)$$

if $n \geq 1$, and this "doubling formula" allows to compute (F_n, F_{n+1}) in a repeated squaring fashion. (Since $F_n \approx \phi_+^n$, by Exercise 3.28, the two multiplications in the last step each take more than $n/64$ word operations.)

3.32 (i) The remainders are $f_{n-2}, f_{n-3}, \ldots, f_1, f_0$, and $\rho_i = 1$ for all i. We have $f_0 = 1$, $f_1 = x$, $f_{n+2} = x f_{n+1} + f_n$ for all $n \in \mathbb{N}$, and $\deg f_n = n$ if $n \geq 1$.

(ii) $F_{n+1} = f_n(1)$ for all $n \in \mathbb{N}$.

(iii) Let $f,g \in \mathbb{Q}[x]$ of degrees $n = \deg f > \deg g \geq 0$. Then the number of division steps in the Euclidean Algorithm for (f,g) is at most n, and this is achieved for $f = f_n$ and $g = f_{n-1}$.

Chapter 4

4.1 ALGORITHM 4.13 Remainder modulo a single precision integer.
Input: A nonnegative multiprecision integer $a = \sum_{0 \leq i \leq n} a_i 2^{64i}$, with $0 \leq a_i < 2^{64}$ for all i, and a single precision integer p with $2^{63} \leq p < 2^{64}$.
Output: a rem p.

1. **if** $a_n < p$ **then** $r \longleftarrow a_n$ **else** $r \longleftarrow a_n - p$
2. **for** $i = n-1, n-2, \ldots, 0$ **do**
   compute $q, r^* \in \{0, \ldots, 2^{64}-1\}$ with $r \cdot 2^{64} + a_i = qp + r^*$ and $r^* < p$
   $r \longleftarrow r^*$
3. **return** r

4.2 (i) $(80/63) \cdot 2^{30} \approx 1.36 \cdot 10^9$; (ii) at most $1.36 \cdot 10^{-8}$.

4.3 (ii) We have $b_i = 2b_{i+1} - 2^m y_{i+m} + y_i$ for $i < n-m-1$. So we first compute a rem p, b_{n-m-1} rem p, and 2^m rem b, at a cost of $O(m)$ word operations, by Exercise 4.1. Then b_i rem p can be computed from b_{i+1} rem p and compared to a rem p in time $O(1)$ for each $i < n-m-1$, or $O(n)$ for all i.

(iii) For a fixed i the error probability is at most $k \cdot 10^{-17}$, and the probability that an error happens for some i is at most $nk \cdot 10^{-17}$. This is at most 0.001 as long as $nk \leq 10^{14}$.

4.7 No.

4.8 (i) $a = 353$; (ii) $a = 777$.

4.9 $t_1 = -x^3 - 2x^2 - 1$, g_2 is not invertible, and hence $\mathbb{Q}[x]/\langle f \rangle$ is not a field.

4.11 (i) $h = x^4 + x^3$ is the modular inverse; (ii) $h = x^3 + x^2 + 1$ satisfies $fh \equiv 0 \bmod g$.

4.13 (i) $f = x^3 + 4x^2 + 4x + 5$; (ii) $x^2 + 6x + 1$.

4.14 (i) The second equivalence is Theorem 4.1. Let $h = \gcd(f,m)$. Then $fg \equiv 0 \bmod m \iff m/h$ divides g, by Exercise 3.16, and the first equivalence follows.

(ii) In the ring $\mathbb{Z}$, 2 is neither a unit nor a zero divisor.

4.15 (ii) $a = 5$: unsolvable; $a = 6$: $x \in \{4, 9, 14\}$; $a = 7$: $x = 12$.

4.16 There are precisely $(q-1)^{\ell+1} q^{n_0}$ pairs of polynomials with degree sequence $(n_0, n_1, \ldots, n_\ell)$ if $n_0 \geq n_1$.

4.17 (i) Let $s = \#S$ and $t = \#T$ for short. The number of pairs f, g of degree n and m, respectively, with a given degree sequence of length ℓ is $(q-1)^{\ell+1}q^n$, by Exercise 4.16. The degree sequences are in one-to-one correspondence with the subsets of $\{0,\ldots,m-1\}$, and hence the number of degree sequences of a given length $\ell \in \{t+1,\ldots,m+1-s\}$ containing no element of S and all elements of T is $\binom{m-s-t}{\ell-1-t}$. The number of all pairs of polynomials of degree n and m, respectively, is $(q-1)^2q^{n+m}$. Thus

$$\begin{aligned} p_{S,T} &= \frac{\displaystyle\sum_{t+1\le\ell\le m+1-s} \binom{m-s-t}{\ell-1-t}(q-1)^{\ell+1}q^n}{(q-1)^2q^{n+m}} \\ &= q^{-m}(q-1)^t \sum_{0\le\ell\le m-s-t} \binom{m-s-t}{\ell}(q-1)^\ell \\ &= q^{-m}(q-1)^t(1+q-1)^{m-s-t} = q^{-s}(1-q^{-1})^t. \end{aligned}$$

(ii) We have $\mathrm{prob}(X_i = 0) = p_{\{i\},\emptyset} = q^{-1}$ and $\mathrm{prob}(X_i = 1) = 1 - \mathrm{prob}(X_i = 0) = 1 - q^{-1}$. The independence follows from

$$\begin{aligned} &\mathrm{prob}(X_i = 0 \text{ for } i \in S \text{ and } X_i = 1 \text{ for } i \in T) \\ &\quad = p_{S,T} = q^{-s}(1-q^{-1})^t = \prod_{i\in S}\mathrm{prob}(X_i = 0)\cdot\prod_{i\in T}\mathrm{prob}(X_i = 1), \end{aligned}$$

for all disjoint subsets $S, T \subseteq \{0,\ldots,m-1\}$.

4.21 ALGORITHM 4.14 Brauer's algorithm.
Input: $a \in R$, where R is ring with 1, $n \in \mathbb{N}_{>0}$, and a parameter $k \in \mathbb{N}_{>0}$.
Output: $a^n \in R$.

1. $q \longleftarrow 2^k$
   let $n = n_l q^l + n_{l-1}q^{l-1} + \cdots + n_1 q + n_0$ with $0 \le n_l,\ldots,n_0 < q$ and $n_l \ne 0$
2. **for** $j = 2,3,\ldots,q-1$ **do** compute a^j
3. $b_l \longleftarrow a^{n_l}$
4. **for** $i = l-1,\ldots,0$ **do** $b_i \longleftarrow b_{i+1}^q \cdot a^{n_i}$
5. **return** b_0

Each execution of the loop body in step 4 takes k squarings and one multiplication. Thus the algorithm uses $k\lfloor(\log n)/k\rfloor \le \log n$ squarings and $w_{2^k}(n) + 2^k - 2 \le (\log n)/k + 2^k - 1$ ordinary multiplications, where $\log = \log_2$ and $w_{2^k}(n)$ is the number of nonzero digits in the 2^k-ary representation of n. Choosing $k = \lfloor \mathrm{loglog}\, n - \mathrm{logloglog}\, n \rfloor$ leads to an overall cost of $(1+o(1))\log n$ (Brauer 1939).

4.24 (i) We proceed by induction on n. The case $n = 1$ is trivial, and we assume that $n > 1$. Let $h^* = \gcd(f_1, \ldots, f_{n-1})$. By the induction hypothesis, there exist $s_1^*, \ldots, s_{n-1}^* \in R$ such that $s_1^* f_1 + \cdots + s_{n-1}^* f_{n-1}^* = h^*$. Now the Extended Euclidean Algorithm computes $s, t \in R$ such that $sh^* + t f_n = \gcd(h^*, f_n) = h$, and the claim follows by letting $s_i = ss_i^*$ for $1 \le i < n$ and $s_n = t$.

(ii) For part (i) of Theorem 4.11, let $h = \gcd(f_1, \ldots, f_n)$. It is clear that if a solution exists, then h divides a, since h divides any linear combination of $f_1, \ldots, f_n$. Conversely, if h divides a, then using the first part of this exercise and multiplying by h/a yields a solution of the linear Diophantine equation.

To prove part (ii), we rewrite (6) in the form $sf^T = a$, where $s = (s_1, \ldots, s_n)$ and $f = (f_1, \ldots, f_n)$ are vectors in R^n. Then for all $s^* \in R^n$, we have

$$s^* f^T = a = sf^T \iff (s^* - s) f^T = 0 \iff s^* - s \in U \iff s^* \in s + U.$$

To show part (iii) of the Theorem, we note that the s_{ij} may be obtained inductively as in the first part of this exercise. The inclusion $Ru_2 + \cdots + Ru_n \subseteq U$ is clear since any u_i is in U and hence any linear combination of the u_i is. Conversely, let $s = (s_1, \ldots, s_n) \in U$. Then $h_{n-1} = \gcd(f_1, \ldots, f_{n-1})$ divides $s_n f_n$, h_{n-1}/h_n divides s_n, and hence $s + q_n u_n \in U$ for $q_n = s_n h_n / h_{n-1}$. Now the last component of that vector is zero, and we conclude inductively that there exist $q_2, \ldots, q_n \in R$ such that $s^* = s + \sum_{2 \le i \le n} q_i u_i \in U$ and has the second up to the nth component zero. Finally, $s^* f^T = 0$ implies that $s^* = 0$, and the claim follows.

(iii) Dividing s_i by l/f_i with remainder if necessary, we may assume that $\deg s_i + \deg f_i < \deg l$ for $2 \le i \le l$. Since $\deg a < l$, this implies that also $\deg s_1 + \deg f_1 < \deg l$.

4.26 (i) $14/3 = [4,1,2]$ and $3/14 = [0,4,1,2]$.

(ii) $[2,1,4] = 14/5$ and $[0,1,1,100] = 101/102$.

4.27 $\sqrt{2} = [2,\overline{1}]$, $\sqrt{2} - 1 = [\overline{1}]$, $\sqrt{2}/2 = [0,\overline{1}]$, $\sqrt{5} = [2,\overline{4}]$, $\sqrt{7} = [2,\overline{1,1,1,4}]$.

4.29 (iii) Induction on i or Exercises 3.20 (iv) and 4.28.

(iv) The first claim is again proven by induction on i. Then

$$g - c_i = -\frac{t_{i-1} - \alpha_i t_i}{s_{i-1} - \alpha_i s_i} + \frac{t_i}{s_i} = -\frac{t_{i-1}s_i - t_i s_{i-1}}{s_i(s_{i-1} - \alpha_i s_i)} = \frac{(-1)^{i+1}}{s_i(s_{i-1} - \alpha_i s_i)},$$

and $\deg(g - c_i) = -\deg s_i - \deg(s_{i-1} - \alpha_i s_i) \le -2\deg s_i - \deg \alpha_i < -2\deg s_i$ if $i \ge 2$.

(v) We have

$$\begin{aligned}\deg r_i &= \deg\left(s_i r_1 \left(\frac{r_0}{r_1} + \frac{t_i}{s_i}\right)\right) = 2n + k + \deg\left(\left(\frac{r_0}{r_1} - g\right) + \left(g + \frac{t_i}{s_i}\right)\right) \\ &< 2n + k + \max\{-n-k, -2n\} \le k\end{aligned}$$

and $r_i/s_i = r_0 + t_i/s_i x^{n+k} \equiv r_0 \bmod x^{n+k}$.

4.32 Up to sign, the f_i and q_i coincide with the remainders and the quotients in the traditional Euclidean Algorithm, and Lemma 3.8 implies that $f_\ell = \gcd(f, f')$ is a constant and f_i and f_{i+1} have no common roots for $0 \le i < \ell$.

Obviously w is constant on all subintervals of (b,c) containing no roots of any f_i, and we only have to investigate what happens immediately to the left and to the right of a zero of some f_i. Thus we may assume that there is some $t \in (b,c)$ such that the two subintervals (b,t) and (t,c) both contain no root of any f_i, and that $f_j(t) = 0$ for some $j \in \{0,\ldots,\ell-1\}$. Suppose first that $j > 0$. Then, as noted above, $f_{j-1}(t) \neq 0 \neq f_{j+1}(t)$, and hence the signs of f_{j-1} and f_{j+1} are constant on the whole interval. Now

$$f_{j-1}(t) = q_j(t) f_j(t) - f_{j+1}(t) = -f_{j+1}(t)$$

implies that f_{j-1} and f_{j+1} have opposite signs on the whole interval, so that there is precisely one sign change in $f_{j-1}(b), f_j(b), f_{j+1}(b)$ and in $f_{j-1}(c), f_j(c), f_{j+1}(c)$, regardless of the signs of $f_j(b)$ and $f_j(c)$. Thus a zero of f_j with $j > 0$ (there may be several such j) does not make the value of w change. On the other hand, if $j = 0$, then $f_1(t) \neq 0$, whence the sign of f_1 is constant on (b,c), but the sign of f_0 left from t is different to its sign right from t. If $f_1(t) = f'(t) > 0$, then f is increasing near t, and we have one sign change in $f_0(b), f_1(b)$ and none in $f_0(c), f_1(c)$. Similarly, if $f_1(t) < 0$, then f is decreasing near t, and again we have one sign change in $f_0(b), f_1(b)$ and none in $f_0(c), f_1(c)$.

Chapter 5

5.1 (ii) $a_0 = 0$, $a_1 = 0$, $a_2 = 1$, $a_3 = 3$.

5.2 For $n \in \mathbb{N}$, let $b_n = \sum_{1 \le i \le n} a_i p^{n-i} \in \mathbb{Z}$ be p^n times the initial segment of the p-adic expansion of length n, with $b_0 = 0$.

(ii) The p-adic expansion terminates if and only if $p^n s/t = b_n \in \mathbb{N}$ for some $n \in \mathbb{N}$, and since $\gcd(s,t) = 1$, this is in turn equivalent to $t \mid p^n$, and also to $t^* = 1$.

(iii) If $k = 0$, then $s/t = b_l \sum_{j \ge 1} p^{-jl} = b_l/(p^l - 1)$, and since s/t is a reduced fraction, we have $t \mid p^l - 1$, or equivalently, $p^l \equiv 1 \bmod t$. Thus $\gcd(p,t) = 1$ and $\mathrm{ord}_t(p) \mid l$. Conversely, if $\gcd(p,t) = 1$, $l^* = \mathrm{ord}_t(p)$, and we let $b = s(p^{l^*} - 1)/t < p^{l^*}$, then $s/t = b \sum_{j \ge 1} p^{-jl^*}$ yields a purely periodic p-adic expansion, so that $l \le l^*$, and finally $l = l^*$.

(iv) k is the least nonnegative integer such that $p^k s/t - b_k$ has a purely periodic expansion, or equivalently, $p^k s/t - b_k = s_k/t_k$ for two coprime integers $s_k < t_k$ with $\gcd(p,t_k) = 1$, by (iii). Since b_k is an integer and s and t are coprime, we have $t_k = t/\gcd(p^k,t)$. With u as in (i), we find that $\gcd(p,t_k) = \gcd(p^{k+1},t)/\gcd(p^k,t) = \gcd(p^{k+1},u)/\gcd(p^k,u)$, and hence $\gcd(p,t_k) = 1$ if and only if $u \mid p^k$. This proves the second claim, and the first one follows from $t_k = t/\gcd(p^k,t) = t/\gcd(p^k,u) = t^*$ and (iii).

5.4 (i) $f = 3x^2 + 3x + 1$.

(ii) The set of all solutions is $f + \langle x(x-1)(x-2)\rangle$, and hence there are 5^{n-2} solutions of degree at most n if $n \geq 2$, namely $\{f + r \cdot x(x-1)(x-2) : r \in \mathbb{F}_5[x] \text{ and } 3 + \deg r \leq n\}$.

5.5 (i) $f = 6x^2 + 5x + 1$.

5.6 A polynomial $f \in \mathbb{F}_5[x]$ satisfies (38) if and only if $f \equiv r \bmod (x-a)$ for all $a \in \mathbb{F}_5$, or equivalently, $f \equiv r \bmod x^5 - x$, since $x^5 - x = \prod_{a \in \mathbb{F}_5}(x-a)$, by Fermat's little theorem. Thus each such f is of the form $f = r + g \cdot (x^5 - x)$ for some $g \in \mathbb{F}_5[x]$, and all $g \in \mathbb{F}_5[x]$ do occur. We have $\deg f = 5 + \deg g \geq 5$ if $g \neq 0$, and $\deg f = \deg r < 5$ if $g = 0$. With the degree constraint $\deg f \leq 5$, we can take precisely the constant polynomials $0, 1, 2, 3, 4 \in \mathbb{F}_5[x]$ for g, and all five solutions are $r, r + (x^5 - x), r + 2(x^5 - x), r + 3(x^5 - x), r + 4(x^5 - x)$. For $\deg f \leq 6$, all polynomials $g \in \mathbb{F}_5[x]$ of degree at most 1 yield a solution, and there are exactly 25 of them.

5.7 (i) The polynomial $l = \sum_{0 \leq i < n} l_i$ has degree less than n and $l(u_i) = 1$ for $0 \leq i < n$. The polynomial $l - 1$ has degree less than n and n roots $u_0, \ldots, u_{n-1}$, hence is the zero polynomial.

5.8 The claim is clear if $u_i = u_j$ for some $i \neq j$, and we may assume that all u_i are distinct. There is nothing to prove if $n = 1$. So we let $n > 1$ and $d = \mathrm{VDM}(u_0, \ldots, u_{n-2}, x) \in R[x]^{n \times n}$. Then $d(u_i) = 0$ for $0 \leq i \leq n-2$, so that d is divisible by $p = (x - u_0) \cdots (x - u_{n-2})$. Now p is monic of degree $n-1$ and $\deg d \leq n-1$, and we conclude that $d = \mathrm{lc}(d)p$. Laplace expansion along the last row shows that $\mathrm{lc}(d) = \mathrm{VDM}(u_0, \ldots, u_{n-2})$, and the claim follows from the induction hypothesis and substituting u_{n-1} for x.

5.11 The cost for computing $g(u_1), \ldots, g(u_{n-1})$ is $3n - 3$ arithmetic operations in F, and the cost for computing $(x - u_0)g + v_0$ after the recursive call is another $2n - 2$ operations. Thus the overall cost is $\sum_{1 \leq i \leq n}(5n - 5) = \frac{5}{2}(n^2 - n)$.

5.12 (i) The polynomial $f(x) - f(-x)$ has degree less than $2n$ and $2n$ zeroes $\pm u_0, \ldots, \pm u_{n-1}$ and hence is the zero polynomial.

(ii) For the existence, we take the Lagrange interpolation polynomial of degree less than $2n$ such that $f(\pm u_i) = v_i$ for all i. By (i), this is an even polynomial. If there is another even polynomial f^* such that $f^*(u_i) = v_i$ for all i, then also $f_*(-u_i) = v_i$ for all i, and the uniqueness follows from the uniqueness of the interpolating polynomial at the $2n$ points $\pm u_0, \ldots, \pm u_{n-1}$.

(iii) $g(x^2) = f(x)$.

(iv) The corresponding statements are:

- If $f \in F[x]$ of degree less than $2n$ is such that $f(-u_i) = -f(u_i)$ for all i, then $f(-x) = -f(x)$.
- There is a unique odd interpolating polynomial $f \in F[x]$ of degree less than $2n$ such that $f(u_i) = v_i$ for all i.

- If $g \in F[x]$ of degree less than n is such that $g(u_i^2) = v_i/u_i$ for all i, then $f(x) = xg(x^2)$.

(v) $f_0 = \dfrac{9(\sqrt{3}-2)}{\pi^2}x^4 - \dfrac{3(5\sqrt{3}-8)}{2\pi}x^2 + \dfrac{3(\sqrt{3}-1)}{2}$, $f_1 = -\dfrac{18(3\sqrt{3}-5)}{\pi^3}x^5 + \dfrac{9(4\sqrt{3}-7)}{\pi^2}x^3 - \dfrac{9\sqrt{3}-22}{2\pi}x$.

5.13 (iii) $f = (5y^2+7y)x^3 + (6y^2+4y+1)x^2 + (9y^2+4y)x + 3y^2 + 7 + 4y$.

5.14 Let $h \in F[x]$ of degree at most $n-2$ such that $h(u_i) = f(u_i)$ for $0 \le i \le n-2$. Then the set of all interpolation polynomials of degree less than n at the $n-2$ points is $\{h + d \cdot (x-u_0)\cdots(x-u_{n-2}) : d \in F\}$ and contains exactly $\#F$ elements if F is finite. Precisely one of them also satisfies $g(0) = c$, since the interpolation problem $g(u_i) = f(u_i)$ for $i \le n-2$ and $g(0) = c$ has exactly one solution of degree less than n. Thus each element $c \in F$ is equally "likely" without the secret of player $n-1$.

5.15 $f = 23$.

5.16 The set of all solutions is $1234 + 2431\mathbb{Z}$, where $2431 = 11 \cdot 13 \cdot 17$, and there are precisely $\lfloor (10^6 - 1 - 1234)/2431 \rfloor = 410$ nonnegative solutions less than 10^6.

5.17 They agreed to meet again on 24 December 1999.

5.18 Sesamy Street is 555 feet long.

5.19 (i) The polynomial $x^2 - 2$ has no roots in $\mathbb{F}_5$, so that it is irreducible. Thus $(x^2-2)^2$ is reducible and has no roots.

(ii) Since all polynomials have degree at most three, it suffices to check that they have no zeroes. This reveals that m_0, m_1 and m_3 are irreducible, while m_2 has the linear factor $x-2$.

(iii) Since m_0 and m_1 are irreducible, monic, and distinct, they are coprime, and the Chinese Remainder Theorem guarantees that a solution exists. Using the Chinese Remainder Algorithm, we find that $f = 3x^3 + 3x^2 + 4x + 4$.

5.20 After multiplying the second congruence with the inverse $-x$ of x modulo x^2+1 and dividing the third congruence (including the modulus) by $x+1$, we obtain the equivalent system

$$f \equiv 1 \bmod (x+1), \quad f \equiv -x+1 \bmod (x^2+1), \quad f \equiv 1 \bmod (x^2+x+1).$$

Its unique solution of least degree is $f = 2x^4 + 3x^3 + 2x + 4$, and the set of all solutions is $f + \langle (x+1)(x^2+1)(x^2+x+1) \rangle$.

5.22 (i) By the Chinese Remainder Theorem, an interpolating polynomial satisfying (39) exists if and only if interpolating polynomials modulo p_0 and modulo p_1 exist. If, for some fixed $k \in \{0,1\}$, we have $u_i \equiv u_j \bmod p_k$ but $v_i \not\equiv v_j \bmod p_k$, then clearly no interpolating polynomial modulo p_k exists. On the other hand, if this is not the case, then we obtain an interpolating polynomial modulo p_k by simply ignoring each duplicate pair (u_i, v_i) modulo p_k.

(ii) Again by the Chinese Remainder Theorem, the interpolating polynomial of degree less than n is unique modulo m if and only if it is unique modulo each p_k, and this in turn is equivalent to saying that all points u_i are distinct modulo each p_k.

(iii) Modulo 3, the first two conditions are equivalent, and the problem reduces to finding a $g \in \mathbb{F}_3[x]$ of degree less than 3 satisfying

$$g(1) \equiv 2 \bmod 3, \quad g(2) \bmod 2 \bmod 3.$$

Obviously $g_0 = 2$ is the solution of least degree, and there are two further solutions of degree less than 3, namely

$$g_1 = g_0 + (x-1)(x-2) = x^2 + 2 \text{ and } g_2 = g_0 + 2(x-1)(x-2) = 2x^2.$$

Modulo 5, we are looking for a polynomial $h \in \mathbb{F}_5[x]$ of degree less than 3 satisfying

$$h(1) \equiv 2 \bmod 5,\ h(2) \equiv 0 \bmod 5,\ h(4) \equiv 4 \bmod 5.$$

There is a unique such polynomial given, for example, by the Lagrange interpolation formula

$$\begin{aligned} h &\equiv 2 \cdot \frac{(x-2)(x-4)}{(1-2)(1-4)} + 0 \cdot \frac{(x-1)(x-4)}{(2-1)(2-4)} + 4 \cdot \frac{(x-1)(x-2)}{(4-1)(4-2)} \\ &\equiv 4(x^2+4x+3) + 4(x^2+2x+2) \equiv 3x^2 + 4x \bmod 5. \end{aligned}$$

Thus there are precisely three interpolating polynomials modulo 15 given by

$$f \equiv g_i \bmod 3,\ f \equiv 3x^2 + 4x \bmod 5$$

for $i = 0,1,2$, and we may compute them by using the Chinese Remainder Algorithm. By inspection, we see that the required modular inverses are given by $2 \cdot 3 + (-1) \cdot 5 = 1$. Then

$$\begin{aligned} f &\equiv g_i \cdot (-1) \cdot 5 + (3x^2+4x) \cdot 2 \cdot 3 \equiv (2 + i(x^2+2)) \cdot 10 + (3x^2+4x) \cdot 6 \\ &\equiv (3+10i)x^2 + 9x + (5+5i) \bmod 15 \end{aligned}$$

for $i = 0,1,2$, and the three solutions are $3x^2+9x+5$, $13x^2+9x+10$, and $8x^2+9x$.

5.23 (i) Let $g = \gcd(m_0, m_1) \in R$, and suppose that $f \in R$ satisfies both congruences. Then $v_0 + s_0 m_0 = c = v_1 + s_1 m_1$ for some $s_0, s_1 \in R$, and hence $v_0 - v_1 = s_1 m_1 - s_0 m_0$. The right hand side is divisible by g, and so is $v_0 - v_1$.

Conversely, we assume that g divides $v_0 - v_1$. By Theorem 4.10, we may compute $s,t \in R$ such that $sm_0 + tm_1 = v_0 - v_1$, using the Extended Euclidean Algorithm, and then $f = v_0 - sm_0 = v_1 + tm_1$ solves the congruences.

(ii) The set of all solutions is $-34 + 252\mathbb{Z}$, since the solution is unique modulo $252 = \operatorname{lcm}(36,42)$.

5.25 Yes, both are isomorphic to $\mathbb{Z}_{60}$.

5.32 (i) The sum formula for the determinant (Section 25.5) gives $r = mn$.

(ii) ALGORITHM 5.32 Small primes modular determinant over $F[x]$.
Input: $A = (a_{ij})_{1\le i,j\le n} \in F[x]^{n\times n}$ with $\deg a_{ij} \le m$ for all i, j, where F is a field with more than mn elements.
Output: $\det A \in F[x]$.

1. $r \longleftarrow mn$, choose $r+1$ distinct points $u_0, \ldots, u_r$ in F
2. **for** $i = 0, \ldots, r$ compute $A(u_i)$
3. **for** $i = 0, \ldots, r$ **do** $d_i \longleftarrow \det A(u_i)$
4. compute $d \in F[x]$ of degree at most r such that $d(u_i) = d_i$ for $0 \le i \le r$ by interpolation
5. **return** d

The cost is $O(m^2n^3)$ for step 2, $O(mn^4)$ for all Gaussian eliminations in step 3, and $O(m^2n^2)$ for step 4, in total $O(mn^4 + m^2n^3)$.

(iii) $\det A = -2x^3 - 2x^2 - 3x$; (iv) $r = m_1 + \cdots + m_n$; (v) $\det A = -2x^2 + 3x + 2$.

5.34 (i) $|c_i| \le \sum_{j+k=i} |a_j b_k| \le nB^2$.

(ii) ALGORITHM 5.33 Small primes modular multiplication in $\mathbb{Z}[x]$.
Input: $a, b \in \mathbb{Z}[x]$ of degree less than n and with max-norm at most B.
Output: $ab \in \mathbb{Z}[x]$.

1. $C \longleftarrow nB^2$, $r \longleftarrow \lceil \log_2(2C+1) \rceil$
   choose r distinct prime numbers $m_0, \ldots, m_{r-1} \in \mathbb{N}$
2. **for** $i = 0, \ldots, r-1$ compute $c_i \in \mathbb{Z}[x]$ of degree at most $2n-2$ and with max-norm at most $m_i/2$ such that $c_i \equiv a_i b_i \bmod m_i$, using polynomial multiplication in $\mathbb{Z}_{m_i}[x]$
3. **call** the Chinese Remainder Algorithm 5.4 to compute $c \in \mathbb{Z}[x]$ of degree at most $2n-2$ and with max-norm at most C such that $d \equiv d_i \bmod m_i$ for $0 \le i < r$
4. **return** c

5.37 $k = 5$: $x^4 + 2x^3 + 2x^2 + x + 1$, $k = 4$: no solution, $k = 3, 2, 1$: $1/(x^2 + x + 1)$.

5.38 Suppose that $r, t \in F[x]$ is a solution. But r has at least $\#S \ge k$ roots and at least $n - \#S > 1$ non-roots, whence $\deg r \ge k$, a contradiction.

5.41 (ii) For $k \in \{n_j + 1, \ldots, n_{j-1}\}$.

5.43 (i) unsolvable; (ii) $\rho = \dfrac{16}{-3x^3 - 7x^2 - 9x + 3}$.

5.45 (i) $\dfrac{1}{4}\dfrac{1}{(x+1)^2} + \dfrac{1}{2}\dfrac{1}{x+1} + \dfrac{3}{4}\dfrac{1}{(x-1)^2} - \dfrac{1}{2}\dfrac{1}{x-1}$; (ii) $-\dfrac{1}{x^3} + \dfrac{2}{x^2} + \dfrac{1}{x} - \dfrac{2}{x^2+1}$.

Chapter 6

6.4 Let $f = (\sum_{0\le i\le n} a_i x^i)/b \in K[x]$ and $c = c^*/d \in K$, with all a_i and c^* in R and $b,d \in R\setminus\{0\}$. Then

$$\operatorname{cont}(f) = \gcd(a_0,\ldots,a_n)/\operatorname{normal}(b), \quad \operatorname{cont}(c) = \operatorname{normal}(c^*)/\operatorname{normal}(d),$$

and

$$\begin{aligned}\operatorname{cont}(cf) &= \operatorname{cont}\left(\frac{\sum_{0\le i\le n} c^* a_i x^i}{bd}\right) = \frac{\gcd(c^*a_0,\ldots,c^*a_n)}{\operatorname{normal}(bd)} \\ &= \frac{\operatorname{normal}(c^*)\gcd(a_0,\ldots,a_n)}{\operatorname{normal}(b)\operatorname{normal}(d)} = \operatorname{cont}(c)\operatorname{cont}(f).\end{aligned}$$

This proves Lemma 6.5. Now let $g = (\sum_{0\le i\le m} c_i x^i)/d \in K[x]$, with all $c_i \in R$ and d as before. We may assume that $fg \neq 0$. Then Lemma 6.5 and Corollary 6.7 (over R) yield

$$\begin{aligned}\operatorname{cont}(bd)\operatorname{cont}(fg) &= \operatorname{cont}(bf\cdot dg) = \operatorname{cont}(bf)\operatorname{cont}(dg) \\ &= \operatorname{cont}(b)\operatorname{cont}(d)\operatorname{cont}(f)\operatorname{cont}(g).\end{aligned}$$

Now $0 \neq \operatorname{cont}(bd) = \operatorname{normal}(bd) = \operatorname{normal}(b)\operatorname{normal}(d) = \operatorname{cont}(b)\operatorname{cont}(d)$ implies that $\operatorname{cont}(fg) = \operatorname{cont}(f)\operatorname{cont}(g)$, and finally

$$\operatorname{pp}(fg) = fg/\operatorname{cont}(fg) = f/\operatorname{cont}(f)\cdot g/\operatorname{cont}(g) = \operatorname{pp}(f)\operatorname{pp}(g).$$

6.5 cont and pp is not a normal form on $K[x]$ since there are associate elements which have different normal forms: for example, we have $\operatorname{pp}(-1) = -1 \neq 1 = \operatorname{pp}(1)$ in $\mathbb{Q}[x]$.

6.8 Let q be a prime divisor of $2A+1$.

6.9 (ii) $R^\times = \{\pm 2^i : i \in \mathbb{Z}\}$.

(iii) Every nonzero element $b \in R$ can be uniquely written as $b = a2^i$ with $a,i \in \mathbb{Z}$ and a odd, and $\operatorname{normal}(b) = |a|$ defines a normal form.

(iv) $\operatorname{cont}_{\mathbb{Z}}(f) = 2$, $\operatorname{cont}_R(f) = \operatorname{cont}_{\mathbb{Q}}(f) = 1$.

6.10 We may assume that $\deg f + \deg g \ge 1$. By Corollary 6.21, there exist polynomials $s,t \in \mathbb{Z}[x]$ such that $sf + tg = r$. Plugging in $x = u$, we have $s(u)f(u) + t(u)g(u) = r$, and since $\gcd(f(u),g(u))$ divides the left hand side of this equation, it divides r.

6.11 The entries S_{ij} of the Sylvester matrix $S = \operatorname{Syl}_y(f,g)$ are

$$S_{ij} = \begin{cases} f_{n-i+j} & \text{if} \quad 1 \le j \le m, \\ g_{j-i} & \text{if} \quad m < j \le m+n. \end{cases}$$

For a typical summand of the determinant, given by a permutation σ on the index set $\{1,2,\ldots,n+m\}$, we have

$$\begin{aligned}\deg_y \prod_{1\le j\le n+m} S_{\sigma_j,j} &= \sum_{1\le j\le m} \deg_y f_{n-\sigma_j+j} + \sum_{m<j\le m+n} \deg_y g_{j-\sigma_j} \\ &\le \sum_{1\le j\le n+m} \sigma_j - \sum_{1\le j\le m} j + \sum_{m<j\le m+n} (m-j) = nm.\end{aligned}$$

6.12 (i) The claims are trivial if $n=0$ or $m=0$, and we may assume that both are positive. Let $f_n = \mathrm{lc}(f)$ and $g_m = \mathrm{lc}(g)$. We start with indeterminates $a_1,\ldots,a_n,b_1,\ldots,b_m$, and let $f^* = f_n\prod_{1\le i\le n}(x-a_i)$ and $g^* = g_m\prod_{1\le j\le m}(x-b_j)$ in the UFD $R[a_1,\ldots,a_n,b_1,\ldots,b_m][x]$. Now we let $i\le n$ and $j\le m$, and denote the homomorphism which substitutes b_j for a_i by a bar. Its kernel is the ideal $\langle a_i-b_j\rangle$. Then $x-b_j$ divides $\gcd(\overline{f^*},\overline{g^*})$, and since $\overline{f_n}=f_n$ and $\overline{g_m}=g_m$ are nonzero, we have $\overline{\mathrm{res}(f^*,g^*)} = \mathrm{res}(\overline{f^*},\overline{g^*}) = 0$, by the proof of Lemma 6.25. Thus a_i-b_j divides $r^* = \mathrm{res}(f^*,g^*)$, and also $r=\prod_{i,j}(a_i-b_j)$ does, since all the linear factors are pairwise coprime. Now the total degree of r in the a_i and b_j is nm, the total degree of r^* is at most nm, and hence r and r^* agree up to some multiplicative constant from R. We have $r^*(0,\ldots,0,1,\ldots,1) = f_n^m\cdot((-1)^m g_m)^n$ and $r(0,\ldots,0,1,\ldots,1)=(-1)^{nm}$, and hence $r^* = f_n^m g_m^n r$. Letting

$$I = \langle a_1-\alpha_1,\ldots,a_n-\alpha_n,b_1-\beta_1,\ldots,b_m-\beta_m\rangle,$$

we find $\mathrm{res}(f,g) = \mathrm{res}(f^* \bmod I, g^* \bmod I) = r^* \bmod I = f_n^m g_m^n r \bmod I$, again by the proof of Lemma 6.25, and the claims follow.

6.15 (i) Let $f^* = \sum_{0\le i\le n} a_i x^i$ and $g^* = \sum_{0\le i\le m} b_i x^i$ be generic polynomials with coefficients in the UFD $S=\mathbb{Z}[a_0,\ldots,a_n,b_0,\ldots,b_m]$, where the a_i and b_i are indeterminates, and $r^* = \mathrm{res}(f^*,g^*)\in S$. Then Corollary 6.21 yields nonzero polynomials $s^*,t^*\in S[x]$ with $\deg_x s^* < m$ and $\deg_x t^* < n$ such that $s^*f^*+t^*g^* = r^*$. Applying the ring homomorphism $\varphi: S\longrightarrow R$ which maps the a_i to the coefficients of f and the b_i to the coefficients of g, we find polynomials $s=\varphi(s^*)$ and $t=\varphi(t^*)$ in $R[x]$ of the required degrees such that $sf+tg=\varphi(r^*)$. Since $\varphi(a_n)$ and $\varphi(b_m)$ are nonzero, we obtain $\mathrm{Syl}(f,g)$ from $\mathrm{Syl}(f^*,g^*)$ by applying φ to each entry, and $\varphi(r^*) = \mathrm{res}(f,g) = r$.

(ii) If r is a unit, then (i) yields $s,t\in R[x]$ of the required degrees with $sf+tg=r$, and hence $(r^{-1}s)f+(r^{-1}t)g=1$. Conversely, let $s_0f+t_0g=1$ for $s_0,t_0\in R[x]$ with $\deg s_0 < m$ and $\deg t_0 < n$. Since f is monic, we find $q\in R$ and $t_1\in R[x]$ such that $xt_0 = qf+t_1$ and $\deg t_1 < n$. So we let $s_1 = xs_0-qg$, and then $s_1f+t_1g = x$. If $n+m\ge 2$, then comparing degrees on both sides yields $\deg(t_1g) = \deg t_1 + m < n+m$, since g is monic, and hence $\deg t_1 < n$. Proceeding inductively, we find polynomials $s_1,s_2,\ldots,s_{n+m-1}$ and $t_1,t_2,\ldots,t_{n+m-1}$ in $R[x]$ with $\deg s_i < m$, $\deg t_i < n$, and $s_if+t_ig = x^i$ for $0\le i<n+m$. Summarizing these $n+m$ equations into one matrix equation, we find that $\mathrm{Syl}(f,g)\cdot A = I$ for a matrix $A\in R^{(n+m)\times(n+m)}$ and the $(n+m)\times(n+m)$ identity matrix I. Taking determinants yields $r\cdot\det A = 1$.

6.18 (i) $\operatorname{res}(f,g) = 184\,140\,000 = 2^5 \cdot 3^3 \cdot 5^4 \cdot 11 \cdot 31$.

(ii) We have $h = 1$. From (i) and Lemma 6.25, we conclude that the gcd of f mod p and g mod p is nonconstant for $p = 2,3,5,11,31$, and 1 for all other primes.

6.21 The stated method is not a probabilistic algorithm because there are specific pairs of inputs (say x and x^2) where it *never* returns the correct output.

6.23 From $f(\alpha) = 0$, we find that

$$1 \le \sum_{0 \le i < n} \left| \frac{f_i}{f_n} \alpha^{i-n} \right| \le \sum_{0 \le i < n} \left(\frac{b}{|\alpha|} \right)^{n-i} \le \sum_{i \ge 1} \left(\frac{b}{|\alpha|} \right)^{i} = \frac{b}{\alpha - b}.$$

6.24 (i) Let $l = n+m-2k$ and $a = \sum_{0 \le i < l} a_i x^i \in R[x]$, with all $a_i \in R$. Then the coefficients of $x^{n+m-1}, x^{n+m-2}, \ldots, x^{2k}$ in the product polynomial ha are precisely the coefficients of $H \cdot (a_{l-1}, \ldots, a_1, a_0)^T$. Since each column of S is the coordinate vector of $x^i f^*$ or $x^i g^*$ for some $i \in \mathbb{N}$, the corresponding column of HS is a shift of an initial segment of the coefficient sequence of $f = hf^*$ or $g = hg^*$, respectively. Moreover, $\det H = h_k^{n+m-2k}$, and hence $\det T = \deg H \cdot \det S = h_k^{n+m-2k} r$.

(ii) By (i), each column of T has 2-norm at most $(n+1)^{1/2}A$, and Hadamard's inequality 16.6 implies that $|r| \le |\det T| \le (n+1)^{n-k} A^{2n-2k}$.

6.25 We have $p \nmid b$ since $p > b$. Let $\alpha = \operatorname{lc}(h) \in \mathbb{Z}$. Theorem 6.26 shows that $\alpha \mid b$, and that $\alpha v \equiv h \bmod p$ if and only if $p \nmid r$. In this case, we have $w \equiv bv \equiv (b/\alpha)h \bmod p$. Now $\|w\|_\infty < p/2$, Corollary 6.33 shows that $\|(b/\alpha)h\|_\infty \le B < p/2$, and hence $w = (b/\alpha)h$ and $\operatorname{pp}(w) = h$ since b and α are normalized. Conversely, if $p \mid r$, then Theorem 6.26 yields $\deg w > \deg h$, and thus $\operatorname{pp}(w) \ne h$.

6.27 (i) Since h is a common divisor in $\mathbb{Z}[x]$ of f and g, $h(u)$ is a common divisor of $f(u)$ and $g(u)$. Every integral root of h divides the constant coefficient of f, so that it is absolutely at most A, and hence $h(u) \ne 0$.

(iii) We have $cw(u) = v(u)w(u) = \operatorname{cont}(v)h(u)$, and (i) implies that $|w(u)| \le \operatorname{cont}(v) \le \|v\|_\infty \le u/2$. Let $w = \operatorname{lc}(w) \prod_{1 \le i \le \deg w} (x - \alpha_i)$, where the α_i are the complex roots of w, with multiplicities. Since $w \mid h \mid f$, each α_i is a root of f, and Exercise 6.23 yields $\alpha_i \le 2A < u/2$ and $|w(u)| \ge |\operatorname{lc}(w)|(u-2A)^{\deg w}$.

6.30 The gcd is $x^2 + 2ax - 2a^2$.

6.31 For $0 \le k \le l$, let $p_{k,l}(w,b)$ denote the probability that at least k balls are white, and $q_l(w,b) = p_{\lceil l/2 \rceil, l}(w,b)$ the probability that at least half of the chosen balls are white. Then $1 = p_{0,l}(w,b) \ge p_{1,l}(w,b) \ge \cdots \ge p_{l,l}(w,b)$. Exchanging the roles of the white and the black balls proves that $p_{k,l}(b,w)$ is the probability that at least k balls are black, and similarly for $q_l(b,w)$. Thus $q_l(w,b) + q_l(b,w) \ge p_{\lceil l/2 \rceil, l}(w,b) + p_{\lfloor l/2 \rfloor + 1, l}(b,w) = 1$, and in particular $q_l(w,w) \ge 1/2$.

For the induction step, let A be the set of all choices containing ball number $w+1$, B the complementary set of all choices not containing ball number $w+1$, and

W the set of all choices containing at least $\lceil l/2 \rceil$ white balls. We have $q_1(w,b) = w/(w+b) \geq 1/2$, and hence we may assume that $l \geq 2$. Then by the formula for the conditional probability, we have

$$\begin{aligned} q_l(w+1,b) &= \operatorname{prob}(W) = \operatorname{prob}_A(W)\operatorname{prob}(A) + \operatorname{prob}_B(W)\operatorname{prob}(B) \\ &= p_{\lceil l/2 \rceil - 1, l-1}(w,b)\operatorname{prob}(A) + p_{\lceil l/2 \rceil, l}(w,b)\operatorname{prob}(B) \\ &\geq p_{\lceil (l-1)/2 \rceil, l-1}(w,b)\operatorname{prob}(A) + p_{\lceil l/2 \rceil, l}(w,b)\operatorname{prob}(B) \\ &= q_{l-1}(w,b)\operatorname{prob}(A) + q_l(w,b)\operatorname{prob}(B) \\ &\geq \frac{1}{2}(\operatorname{prob}(A) + \operatorname{prob}(B)) = \frac{1}{2}, \end{aligned}$$

by the induction hypothesis, and the claim follows.

6.32 (i) We have $t = \lceil \log_q((4n+2)d) \rceil \in O(\log(nd))$. Example 6.19 shows that the gcd of f and g over $\mathbb{F}_{q^t}$ is the same as over $\mathbb{F}_q$. Algorithm 6.36 takes $O(nd(n+d))$ operations in $\mathbb{F}_{q^t}$, by Theorem 6.37, and one arithmetic operation in $\mathbb{F}_{q^t}$ costs $O(t^2)$ operations in $\mathbb{F}_q$, by Corollary 4.6.

6.33 The six intersection points are $(-3,-1)$, $(-2,11)$, $(-1/2,67/8)$, $(1,-1)$, $(3/2,-17/8)$, and $(3,11)$.

6.34 The minimal polynomial over $\mathbb{Q}$ is $x^4 - 10x^2 + 1$. Over $\mathbb{F}_{19}$, the minimal polynomial of $\alpha + 7\alpha = 8\alpha$ is $x^2 + 4$, and $x^4 - 10x^2 + 1 \equiv (x^2+4)(x^2+5) \bmod 19$.

6.35 (ii) Let $\alpha = \alpha_1, \ldots, \alpha_n \in \mathbb{C}$ be the roots of f. Exercise 6.12 shows that $r = \prod_{1 \leq i \leq n} g(x/\alpha_i)$.

(iii) Take $\operatorname{res}_y(f(y), g((x-ay)/b))$ and $\operatorname{res}_y(g(y), f(xy))$, respectively.

(iv) The minimal polynomial of $\sqrt{2} - 2\sqrt{3}$ over $\mathbb{Q}$ is $x^4 - 28x^2 + 100$, and the minimal polynomial of $\sqrt{2}\sqrt[3]{3}$ is $x^6 - 72$ over both fields.

6.36 (i) By Exercise 6.12, we have $r = \prod_{1 \leq i \leq n}(x - g(\alpha_i))$, where $\alpha = \alpha_1, \ldots, \alpha_n$ are the roots of f in $\mathbb{C}$.

(ii) The minimal polynomials are $x^2 - 2x - 2$ and $x^3 - 3x^2 - 3x - 1$.

6.38 In step 3, use the EEA to compute $s_1, t \in F[x]$ of degree less than d such that $s_1 f + tg = \gcd(f_1, g)$, and set $s_2 = t$ and $s_i = a_i t$ for $3 \leq i \leq n$. The additional cost is $(n-2)d + O(d^2)$ operations in F.

6.39 We call Algorithm 6.45 with the g_i as input, and then divide m by the result. If the division is not possible, then the random choice was unlucky, and we return "FAILURE". If $\deg f_i \leq d$ for all i, then the cost is $O(n^2 d^2)$ for computing m and all g_i, by Theorem 5.7. We have $\deg g_i \leq (n-1)d$ for all i, and the call to Algorithm 6.45 takes $O(n^2 d^2)$ operations, by Theorem 6.46. The same bound is valid for the final division, and hence the overall cost is $O(n^2 d^2)$. Using the fast algorithms from Part II, the cost drops to $O^{\sim}(n^2 d)$. By Theorem 6.46, the error probability is at most $1/2$ if we let $\#S \geq 2(n-1)d$.

6.41 The only nonzero entries in the first row of $\mathrm{Syl}(f,g)$ and its submatrices S_k are $\mathrm{lc}(f)$ and $\mathrm{lc}(g)$, so that this row is divisible by the gcd of the leading coefficients.

6.44 (i) follows by induction on i.

(iii) Similarly to (i), induction on i shows that $q_i \geq A(B+C)^{k-i}C^i$ and all coefficients of r_i are greater or equal to $A(B+C)^{k+1-i}C^i$.

(iv) Let $n = \deg_x a \geq m = \deg_x b$. Then the statement analogous to (i) says that $\deg_y a = \alpha$, $\deg_y b = \beta$, and $\deg_y c = \gamma$ imply $\deg_y q_i \leq \alpha + (k-i)\beta + i\gamma$, $\deg_y r_i \leq \alpha + (k+1-i)\beta + i\gamma$, and $\deg_y r \leq \alpha + (k+1)\beta$. The cost for the pseudodivision is $O(mk^2d^2)$ operations in F.

(In the 1999 edition, the variables are named differently.)

6.46 (i) We have $\rho \geq d$ if and only if $X_j = X_{j+1} = \cdots = X_{j+d-1} = 0$ for some $j \leq m-d$. The probability for this to happen is q^{-d} when j is fixed, and hence $\mathrm{prob}(\rho \geq d) \leq (m-d+1)q^{-d}$. We have $\mathrm{prob}(\rho \geq 0) = 1$, and therefore

$$\mathcal{E}(\rho) \leq 1 + m \sum_{1 \leq d \leq m} q^{-d} \leq 1 + \frac{mq^{-1}}{1-q^{-1}} = 1 + \frac{m}{q-1}.$$

(ii) We define Bernoulli random variables X_i such that $X_i = 1$ if i occurs in the degree sequence and $X_i = 0$ otherwise, for $0 \leq i < m$. Then $\delta = \rho + 1$ if $g \nmid f$, by Exercise 4.17.

(iii) Apply (ii) to q being a prime divisor of $2A+1$.

6.47 (i) Let $\alpha_i^* = \mathrm{lc}(r_i^*)$ for $2 \leq i \leq \ell$. We first prove by induction on i that $r_i^* = \alpha_i^* r_i$ for $0 \leq i \leq \ell$. The start of the induction follows from $r_0^* = f, r_1^* = g$. From the induction hypothesis, we find for $i \geq 1$

$$\begin{aligned}(\alpha_{i-1}^*)^{-1} r_{i+1}^* &= (\alpha_{i-1}^*)^{-1}(r_{i-1}^* - q_i^* r_i^*) = (\alpha_{i-1}^*)^{-1}(\alpha_{i-1}^* r_{i-1} - q_i^* \alpha_i^* r_i) \\ &= r_{i-1} - (\alpha_i^* (\alpha_{i-1}^*)^{-1} q_i^*) r_i, \\ \deg((\alpha_{i-1}^*)^{-1} r_{i+1}^*) &< \deg r_i^* = \deg r_i.\end{aligned}$$

The uniqueness of remainder and quotient on division of r_{i-1} by r_i implies that

$$\rho_{i+1} r_{i+1} = (\alpha_{i-1}^*)^{-1} r_{i+1}^* \text{ and } q_i = \alpha_i^* (\alpha_{i-1}^*)^{-1} q_i^*. \tag{14}$$

In particular, r_{i+1}^* is a scalar multiple of r_{i+1}, and hence equal to $\alpha_{i+1}^* r_{i+1}$. Furthermore, the lengths of the two algorithms are equal.

Comparing leading coefficients in (14), we have $\rho_{i+1} = (\alpha_{i-1}^*)^{-1}\alpha_{i+1}^*$. Inductively, we find $\alpha_i^* = \alpha_i$. Together with (14) this proves the first two claimed equations. The other two follow similarly by induction, or, alternatively, by the uniqueness property in Lemma 5.15.

(ii) For a polynomial $a \in \mathbb{Q}[x]$, we write $\beta(a)$ for the maximal absolute value of the integers that occur in the (relatively prime) numerator and denominator of any coefficient of a. Then we have for $1 \le i \le \ell$:

$$\begin{aligned}
\beta(\alpha_i) &\le \beta(\rho_i)\beta(\rho_{i-2})\cdots \le AC^{\lfloor i/2 \rfloor},\\
\beta(q_i^*) &\le \beta(\alpha_{i-1})\beta(\alpha_i)\beta(q_i) \le \beta(\rho_i)\beta(\rho_{i-1})\cdots\beta(\rho_1)\beta(\rho_0)\beta(q_i) \le A^2C^i,\\
\beta(r_i^*) &\le \beta(\alpha_i)\beta(r_i) \le AC^{\lfloor i/2 \rfloor}\cdot B,\\
\beta(s_i^*) &\le \beta(\alpha_i)\beta(s_i) \le AC^{\lfloor i/2 \rfloor}\cdot B,\\
\beta(t_i^*) &\le \beta(\alpha_i)\beta(t_i) \le AC^{\lfloor i/2 \rfloor}\cdot B.
\end{aligned}$$

Thus all integers are absolutely at most $A^2C^\ell \le C^{m+2}$.

6.48 (i) Let $2 \le i \le \ell$, $n_i = \deg r_i$, $\sigma = \sigma_{n_i}$. As in the proof of Theorem 6.52, we find that $\sigma r_i, \sigma s_i, \sigma t_i$ are in $F[x,y]$. Cramer's rule shows that $\deg_y \sigma \le (n+m-2n_i)d$ and the degree in y of $\sigma r_i, \sigma s_i, \sigma t_i$ is at most $(n+m-2n_i-1)d$.

To bound the degree of the quotients, we consider the pseudodivision (13) on page 181 (in the 2003 edition), as in the proof of Theorem 6.52, and Exercise 6.44 implies that $\deg_y(\sigma_{n_i}^k \sigma_{n_{i-1}} q_i) \le (k+1)(n+m)d$ and $\deg_y(\sigma_{n_i}^{k+1}\sigma_{n_{i-1}}\rho_{i+1}r_{i+1}) \le (k+2)(n+m)d$.

Thus the degree in y of all numerators and denominators in the EEA is at most $(\delta+2)(n+m)d$, and one arithmetic operation on such a coefficient takes $O((n\delta d)^2)$ word operations. Now the claim follows since the number of arithmetic operations is $O(nm)$, by Theorem 3.16 (Theorem 3.11 in the 1999 edition).

(ii) Let $q_i^*, r_i^*, s_i^*, t_i^* \in F(y)[x]$ denote the results of the traditional EEA, $\alpha_i \in F(y)$ as in Theorem 6.53, and $c = (\delta+2)(n+m)d$. Moreover, for $a \in F(y)[x]$, let $\beta(a)$ be the maximal degree in y of the relative prime numerator and denominator of any coefficient of a. Then essentially the same proof as in Exercise 6.47 shows that Theorem 6.53 (i) holds, $\beta(\alpha_i) \le d + c\lceil i/2 \rceil$, and $\beta(q_i^*), \beta(r_i^*), \beta(s_i^*), \beta(t_i^*)$ are all at most $2d + ic \le (m+2)c$. As in (i), the claim now follows from from Theorem 3.11 (Theorem 3.11 in the 1999 edition).

6.49 (i) We have $s_2 = (\rho_0\rho_2)^{-1} = \alpha_2^{-1}$, and hence $\alpha_2^{-1} = \kappa_2 = \gamma_2/\sigma_{n_2}$. By Lemma 3.15 (v), we have

$$s_i t_{i+1} - t_i s_{i+1} = (-1)^i(\rho_0\cdots\rho_{i+1})^{-1} = (-1)^i \alpha_i^{-1}\alpha_{i+1}^{-1}$$

for $i \ge 0$. Comparing constant coefficients, we find that

$$\frac{\gamma_{i+1}}{\sigma_{n_i}\sigma_{n_{i+1}}} = \kappa_i\lambda_{i+1} - \lambda_i\kappa_{i+1} = \frac{(-1)^i}{\alpha_i\alpha_{i+1}}.$$

This yields the first claim, and the second one follows by induction on i.

(ii) By Hadamard's inequality 16.6, we have $|\det Y_j|, |\det Z_j|, |\sigma_{n_j}| \le B$, $|\gamma_2| \le B$, and $|\gamma_j| \le 2B^2$ for $2 \le j \le \ell$. All γ_j are integers, and by (i), there are at most

$\lceil i/2 \rceil$ of them in the denominator of α_i, so that this denominator is at most $(2B)^i$ in absolute value. Similarly, the numerator of α_i contains at most $(i-1)/2$ of the γ_j if i is even, and it contains γ_2 and at most $(i/2)-1$ of the γ_j with $j \geq 3$ if i is odd. Thus the absolute value of the numerator is no more than $(2B)^i$ as well.

For $a \in \mathbb{Q}[x]$, we denote by $\beta(a)$ the maximal absolute value of the coprime numerator and denominator in any coefficient of a, as in Exercise 6.47. Then $\beta(r_i), \beta(s_i), \beta(t_i) \leq B$, by Theorem 6.52, and together with what we have just shown, we find that $\beta(r_i^*), \beta(s_i^*), \beta(t_i^*) \leq (2B)^{i+1}$. The length estimate follows from $i \leq m+1$ and $\log B \in O(n \log(nA))$.

(iii) This is completely analogous to (ii). Let $b = (n+m)d$. The degree in y of $\det Y_j$, $\det Z_j$, and σ_{n_j} is at most b, and $\deg_y \gamma_j \leq 2b$, for $2 \leq j \leq \ell$. Let $\beta(a)$ be the maximal degree in y of the coprime numerators and denominators of any coefficient of $a \in F(y)[x]$. Then $\beta(\alpha_i) \leq ib$, as in (ii), Theorem 6.54 says that $\beta(r_i), \beta(s_i), \beta(t_i) \leq b$, and the claim follows.

6.50 The proof is analogous to the proof of Theorem 6.58. Let $\ell^* \in \mathbb{N}$ be the number of division steps of the Euclidean Algorithm in $F(y)[x]$ of f, g, let $\sigma_{m-1}, \ldots, \sigma_0 \in F[y]$ be their subresultants, and $n_0^* \geq \cdots \geq n_{\ell^*}^* \in \mathbb{N}$ be their degree sequence. Now $\deg_y p > d$, so that p divides none of the leading coefficients of f and g. For any $k \in \{0, \ldots, m\}$, we have $\deg_y p > (n+m)d \geq \deg_y \sigma_k$, whence $p \nmid \sigma_k$ and n_k^* occurs as a remainder degree in the EEA of f mod p and g mod p, by Theorem 6.55. Thus $n_i = n_i^*$ for $0 \leq i \leq \ell = \ell^*$. The numerators and denominators of the coefficients of the $r_i, s_i, t_i \in F(y)[x]$ have degree in y at most $(n+m)d$, by Theorem 6.54, and Theorem 5.16 implies that they can indeed be reconstructed from their images modulo p.

The cost for evaluating f and g at all points in S in step 2 is $O(n^2 d^2)$ operations in F. The cost of the EEA for $f(x,u)$ and $g(x,u)$ is $O(nm)$ field operations per evaluation point u, in total $O(n^2 md)$ field operations. The dominant cost occurs for the Cauchy interpolation in step 3. By Theorem 6.54, the degree in y of both the numerator and the denominator of any coefficient of r_i, s_i, or t_i is at most $(n+m-2n_i)d$, for $0 \leq i \leq \ell$. Thus by Theorem 5.16, $\nu_i = 2(n+m-2n_i)d+1$ points in S_i are sufficient to reconstruct such a coefficient. The cost for computing an interpolating polynomial in $F[y]$ of degree less than ν_i at the ν_i points is $\frac{5}{2}\nu_i^2 + O(\nu_i)$ field operations, by Exercise 5.11, and the rational function reconstruction takes at most another $O(\nu_i^2)$ operations, by Theorem 3.16 (Theorem 3.11 in the 1999 edition). We have $\nu_i \in O(nd)$, there are $O(nm)$ coefficients in total, and hence the overall cost is $O(n^3 md^2)$ operations in F.

In the normal case, we have $n_{\ell-i} = i$ and $\nu_{\ell-i} = 2(n+m-2i)d+1$, and the number of coefficients in $F(y)$ of $r_{\ell-i}, s_{\ell-i}, t_{\ell-i}$ (without the leading coefficient of $r_{\ell-i}$, which is 1) is $n_{\ell-i} + m - n_{\ell-i-1} + 1 + n - n_{\ell-i-1} + 1 = n+m-i+1$, by Lemma 3.15 (b) (Lemma 3.10 in the 1999 edition) for $0 \leq i < m$. Thus the overall

cost for step 3 is

$$5 \sum_{0 \le i < m} (n+m-i+1) \cdot \Big((2(n+m-2i)d+1)^2 + O(nd) \Big)$$

field operations, and a routine calculation (best done with a computer algebra system) shows that this sum can be bounded by $\frac{140}{3} n^3 m d^2 + O(n^3 d(n+d))$.

6.51 ALGORITHM 6.63 Modular EEA in $\mathbb{Q}[x]$: big prime version.
Input: $f, g \in \mathbb{Z}[x]$ with $\deg f = n \ge \deg g = m \ge 1$ and $\|f\|_\infty, \|g\|_\infty \le A$.
Output: The results r_i, s_i, t_i in $\mathbb{Q}[x]$ of the EEA for f and g.

1. $B \longleftarrow (n+1)^n A^{n+m}$
   choose a prime $p \in \mathbb{N}$ with $2B^2 < p < 4B^2$
2. **call** the Euclidean Algorithm 3.14 to compute all results in $\mathbb{Z}_p[x]$ of the EEA for f mod p and g mod p
3. Let $n_0 = n \ge n_1 = m > n_2 > \ldots > n_\ell \ge 0$ be the degrees of all remainders that were computed in step 2
   for $i = 2, \ldots, \ell$ **do**
   compute the coefficients of the monic remainder $r_i \in \mathbb{Q}[x]$ of degree n_i and of $s_i, t_i \in \mathbb{Q}[x]$ from their images modulo p by rational number reconstruction (Section 5.10)
4. **return** r_i, s_i, t_i for $2 \le i \le \ell$

For the modular EEA in $F[x,y]$, we replace $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{Z}_p$ throughout by $F[y]$, $F(y)$, $F[y]/\langle p \rangle$, respectively. The input then are two polynomials $f, g \in F[x,y]$ with $\deg_x f = n \ge \deg_x g = m$ and $\deg_y f, \deg_y g \le d$. In step 1, we choose a monic irreducible polynomial $p \in F[y]$ of degree $2(n+m)d+1$, and we use rational function reconstruction (Section 5.7) instead of rational number reconstruction in step 3.

For $\mathbb{Z}[x]$, the cost for step 2 is $O(nm)$ arithmetic operations in $\mathbb{Z}_p$, each taking $O(\log^2 B)$ or $O(n^2 \log^2(nA))$ word operations. Step 3 takes $O(\log^2 B)$ word operations for each coefficient of some r_i, s_i, or t_i, by Corollary 5.17. There are $O(nm)$ coefficients, and the overall cost is $O(n^3 m \log^2(nA))$ word operations. Similarly, the cost for step 2 in the bivariate case is $O(nm)$ arithmetic operations in the residue class field $F[y]/\langle p \rangle$. Each such operation takes $O(n^2 d^2)$ operations in F. Thus step 2 takes $O(n^3 m d^2)$ operations in F, and the same estimate holds for step 3.

6.53 (i) Let $b = (n+m)d$ and $1 \le i \le \ell$. As in the proof of Theorem 6.62, we find that $\alpha_i \in F[y]$ divides the subresultant $\sigma_{n_i} \in F[y]$ of f and g, and Theorem 6.54 implies that $\deg_y \alpha_i, \deg_y r_i \le b$. Then $\deg_y a_{i-1} \le (\delta+2)b$, and Exercise 6.44 shows that $\deg_y q_i, \deg_y (a_{i-1} \text{ rem } r_i) \le (\delta+2)b$ as well. Thus the degree in y of all coefficients in $F[y]$ throughout the algorithm is at most $(\delta+2)b \in O(n\delta d)$, the cost for one arithmetic operation on such coefficients is $O(n^2 \delta^2 d^2)$ operations in F, and the claim follows since there are $O(nm)$ of them.

(ii) We initialize $s_0 = t_1 = 1$ and $s_1 = t_0 = 0$ in step 1, and additionally compute

$$\rho_{i+1} = \mathrm{cont}_x(a_{i-1} \text{ rem } r_i) \in R, \quad s_{i+1} = (\mathrm{lc}(r_i)^{1+n_{i-1}-n_i} s_{i-1} - q_i s_i)/\rho_{i+1},$$

$$t_{i+1} = (\mathrm{lc}(r_i)^{1+n_{i-1}-n_i} t_{i-1} - q_i t_i)/\rho_{i+1}$$

in the body of the **while** loop in step 2. In general, s_i and t_i need not lie in $R[x]$, but their denominators divide σ_{n_i}/α_i. We obtain from Theorem 6.52 and Theorem 6.54 that their numerators have max-norm at most $(n+1)^n A^{n+m}$ if $R = \mathbb{Z}$ and degree in y at most $(n+m)d$ if $R = F[y]$, respectively, and the same bounds are valid for the denominators.

Let $1 \le i \le \ell$. If $R = \mathbb{Z}$, then

$$|\rho_{i+1}| \le \|a_{i-1} \text{ rem } r_i\|_\infty \le (2B)^{\delta+2}, \quad \|\mathrm{lc}(r_i)^{1+n_{i-1}-n_i} \sigma_{n_{i-1}} s_{i-1}/\alpha_{i-1}\|_\infty \le B^{\delta+2},$$

$$\|q_i \sigma_{n_i} s_i/\alpha_i\|_\infty \le n(2B)^{\delta+3},$$

so that

$$\|(\sigma_{n_{i-1}}\sigma_{n_i}/\alpha_{i-1}\alpha_i)(\mathrm{lc}(r_i)^{1+n_{i-1}-n_i} s_{i-1} - q_i s_i)\|_\infty \le (n+1)(2B)^{\delta+4}.$$

The latter quantity bounds the absolute value of all numerators and denominators in $\mathbb{Z}$ occurring in the algorithm, and hence their length is $O(\delta \log B)$. This yields the same time bound as in Theorem 6.62. The case $R = F[y]$ goes analogously.

6.54 Let $\delta_i = n_{i-1} - n_i$. Assuming that the degree bound from Exercise 6.44 is an equality, we find that

$$\begin{aligned}
\deg_y(a_{i-1} \text{ rem } r_i) &= \deg_y r_{i-1} + (\delta_i + 1)\deg_y r_i \\
&= (n+m-2n_{i-1})d + (\delta_i+1)(n+m-2n_i)d, \\
\deg_y \mathrm{cont}_x(a_{i-1} \text{ rem } r_i) &= \deg_y(a_{i-1} \text{ rem } r_i) - \deg r_{i+1} \\
&= (n+m-2n_{i-1})d + (\delta_i+1)(n+m-2n_i)d \\
&\quad -(n+m-2n_{i+1})d \\
&= -2(\delta_i+\delta_{i+1})d + (\delta_i+1)(n+m-2n_i)d.
\end{aligned}$$

Chapter 7

7.3 The roots of x^3+x+1 are β, β^2, β^4, the roots of x^3+x^2+1 are β^3, β^6, and $\beta^{12} = \beta^5$, and the root of $x+1$ is $\beta^7 = 1$. The following table gives all possible BCH-codes.

δ	generator polynomial g	exponents i with $g(\beta^i) = 0$	$\dim C$	$d(C)$
1	1	Ø	7	1
2,3	x^3+x+1	1,2,4	4	3
4,5,6,7	$x^6+\cdots+x+1$	1,2,3,4,5,6	1	7

7.4 Under the assumption that at most one error has occurred, the transmitted words are

$$c_1 = r_1 = (x^3+x^2+x+1)g \bmod x^7-1,$$
$$c_2 = x^6+x^4+x+1 \bmod x^7-1 = (x^3+1)g \bmod x^7-1.$$

There are precisely three codewords with Hamming distance 2 from r_2, namely $g \bmod x^7-1$,

$$x^6+x^2+1 \bmod x^7-1 = (x^3+x+1)g \bmod x^7-1,$$
$$x^6+x^5+x \bmod x^7-1 = (x^3+x^2+x)g \bmod x^7-1.$$

7.5 (ii) $x^{10}-1$ has the roots $1,\beta,\beta^2,\ldots,\beta^9$.

(iii) For $1 \le \delta < 10$, the polynomial $g_\delta = (x-\beta)(x-\beta^2)\cdots(x-\beta^{\delta-1}) \in \mathbb{F}_{11}[x]$ of degree $\delta-1$ generates a BCH$(11,10,\delta)$ code of dimension $11-\delta$ and minimal distance δ.

(iv) The transmitted word is

$$x^6+5x^3+8x^2+7x+4 \bmod x^{10}-1 = (x^2+8x+4)g \bmod x^{10}-1.$$

Chapter 8

8.1 The following scheme uses three multiplications and three divisions in $\mathbb{R}$ if $b_1 \neq 0$.

$$\frac{a_0+a_1 i}{b_0+b_1 i} = \frac{(a_0+a_1 i)(b_0-b_1 i)}{b_0^2+b_1^2} = \frac{a_0b_0+a_1b_1+(a_1b_0-a_0b_1)i}{b_0^2+b_1^2}$$
$$= \frac{a_0\frac{b_0}{b_1}+a_1}{b_0\frac{b_0}{b_1}+b_1} + \frac{a_1\frac{b_0}{b_1}-a_0}{b_0\frac{b_0}{b_1}+b_1}i$$

A similar scheme works when $b_0 \neq 0$. Lickteig (1987) shows that this is optimal: any such division algorithm uses at least six real multiplications and divisions.

8.4 Induction on k reveals that $9\cdot 3^k - 8\cdot 2^k$ is strictly smaller than $2\cdot 4^k - 2\cdot 2^k + 1$ precisely when $k \ge 5$. Thus in theory, Karatsuba's algorithm is faster than classical multiplication for degrees above $2^5 = 32$.

8.5 When $n = 2^k$, both variants yield the same running time bound $9n^{\log 3} + O(n)$, while the bound for $n = 2^{k-1}+1$ is $27n^{\log 3} + O(n)$ for variant (i) and only $18n^{\log 3} + O(n)$ for variant (ii).

8.6 From Section 2.3, we know that classical multiplication of two polynomials of degrees less than 2^d takes $2\cdot 2^{2d} - 2\cdot 2^d + 1$ operations in R. This yields $\gamma(d) = (2\cdot 2^{2d} + 6\cdot 2^d + 1)/3^d$. Taking derivatives with respect to d, the unique positive $d \in \mathbb{R}$ that minimizes $\gamma(d)$ is $d \approx 2.214$. Of the nearest integers 2 and 3, $d = 2$ yields the smaller value $\gamma(2) = 19/3$. Thus if classical multiplication is used for polynomials of degree less than $2^2 = 4$, then the overall cost is at most $\frac{19}{3}n^{\log 3} + O(n)$, which is significantly smaller than the $9n^{\log 3}$ from Theorem 8.3.

8.7 (i) Proceeding à la Karatsuba, we first compute the products F_0G_0, F_1G_1, F_2G_2, $(F_0+F_1)(G_0+G_1)$, $(F_0+F_2)(G_0+G_2)$, and $(F_1+F_2)(G_1+G_2)$, and obtain

$$\begin{aligned} H_0 &= F_0G_0, \quad H_1 = (F_0+F_1)(G_0+G_1) - F_0G_0 - F_1G_1, \\ H_2 &= (F_0+F_2)(G_0+G_2) - F_0G_0 - F_2G_2 + F_1G_1, \\ H_3 &= (F_1+F_2)(G_1+G_2) - F_1G_1 - F_2G_2, \quad H_4 = F_2G_2. \end{aligned}$$

This leads to an $O(n^{\log_3 6})$ multiplication algorithm.

(ii) The cost of the algorithm is $O(n^{\log_m d})$. This is asymptotically faster than Karatsuba's algorithm if $\log_m d < \log 3$, or equivalently, $d < m^{\log 3} = 3^{\log m}$. For (i), we have $d = 6 > 3^{\log 3} \approx 5.7$, that is, the algorithm is slower than Karatsuba's.

8.8 (i) Correctness follows from noting that P_i is $\alpha(u_i)$ with its coefficients substituted by the F_i, and similarly for Q_i. The number of operations in F used in step 1 is mk multiplications and $(m-1)k$ additions for the computation of each P_i and each Q_i, in total $(4m^2-2m)k$ multiplications and $(4m^2-6m+2)k$ additions. The degree of each R_i is less than $2k-1$, and the cost for step 3 is $(2m-1)(2k-1)$ multiplications and $(2m-2)(2k-1)$ additions per H_i, in total $(2m-1)^2(2k-1)$ additions and $(4m^2-3m-2)(2k-1)$ multiplications. Thus the overall cost is about $12m^2k$ or $12mn$ multiplications and approximately the same number of additions.

(ii) We have

$$(u_i^j)_{0\le i,j<5} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 & 1 \\ 1 & 3 & 4 & 2 & 1 \\ 1 & 4 & 1 & 4 & 1 \end{pmatrix}, \quad (c_{ij})_{0\le i,j<5} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 2 & 3 & 1 \\ 0 & 4 & 1 & 1 & 4 \\ 0 & 4 & 3 & 2 & 1 \\ 4 & 4 & 4 & 4 & 4 \end{pmatrix}.$$

This leads to the following scheme for computing $H_0, \ldots, H_4$. First, we compute

$$\begin{aligned} P_0 &= F_0G_0, \\ P_1 &= (F_0+F_1+F_2)(G_0+G_2+G_3), \\ P_2 &= (F_0+2F_1+4F_2)(G_0+2G_1+4G_2), \\ P_3 &= (F_0+3F_1+4F_2)(G_0+3G_1+4G_2), \\ P_4 &= (F_0+4F_1+F_2)(G_0+4G_1+G_2). \end{aligned}$$

Then we have $R_i = P_i Q_i$ for all i, and finally

$$\begin{aligned} H_0 &= R_0, \\ H_1 &= 4R_1 + 2R_2 + 3R_3 + R_4, \\ H_2 &= 4R_1 + R_2 + R_3 + 4R_4, \\ H_3 &= 4R_1 + 3R_2 + 2R_3 + R_4, \\ H_4 &= 4R_0 + 4R_1 + 4R_2 + 4R_3 + 4R_4. \end{aligned}$$

(iii) As in Exercise 8.7, the cost of the recursive algorithm is $O(n^{\log_m(2m+1)})$, and the claim follows since $\lim_{m\longrightarrow\infty} \log_m(2m+1) = 1$.

8.13 (i) We have $(\omega^{-1})^n = (\omega^n)^{-1} = 1$, and $(\omega^{-1})^\ell - 1 = \omega^{n-\ell} - 1$ is not a zero divisor for $1 \leq \ell < n$, by Lemma 8.7.

(iii) Let $k = e \cdot \gcd(n,k)$, with $e \in \mathbb{N}$. Then $(\omega^k)^d = (\omega^n)^e = 1$, and ω^k is a dth root of unity. Since n is a unit, so is d, with inverse $n^{-1}\gcd(n,k)$. Finally, let $\ell \in \{1,\ldots,d-1\}$. Division with remainder yields $q,r \in \mathbb{N}$ such that $\ell k = qn + r$, with $0 < r < n$ since ℓk is not divisible by n. Thus $(\omega^k)^\ell - 1 = (\omega^n)^q \omega^r - 1 = \omega^r - 1$ is not a zero divisor, again by Lemma 8.7, and the claim follows.

8.14 η is a primitive $2n$th root of unity if and only if 2 is a unit in R.

8.15 (i) For $\omega, \eta \in R_n$, we have $(\omega\eta)^n = \omega^n \eta^n = 1$ and $(\omega^{-1})^n = (\omega^n)^{-1} = 1$.

(ii) Since a field contains no nonzero zero divisors, (a) and (b) are clearly equivalent, and the implications (b) $\Longrightarrow$ (c) $\Longrightarrow$ (d) are obvious. To prove (d) $\Longrightarrow$ (b), we let $\ell \in \{1,\ldots,n-1\}$. Using the Extended Euclidean Algorithm, we find $s,t \in \mathbb{Z}$ such that $sn + t\ell = \gcd(n,\ell)$. Now $\gcd(n,\ell) < n$, and hence there is some prime divisor p of n such that $\gcd(n,\ell)$ divides n/p. If we let $k \in \mathbb{N}$ such that $k \cdot \gcd(n,\ell) = n/p$, then $(\omega^\ell)^{kt} = \omega^{ksn+kt\ell} = \omega^{n/p} \neq 1$, and we conclude that also $\omega^\ell \neq 1$.

(iv) The map $\varphi: n \longmapsto \omega^n$ from $\mathbb{Z}_n$ to R_n is a group homomorphism. Since ω is a primitive nth root of unity, φ is injective, and hence $\#R_n \geq n$. On the other hand, each element of R_n is a root of the polynomial $x^n - 1$, which has at most n roots in the integral domain R. Thus $\#R_n \leq n$ as well, and we conclude that $\#R_n = n$ and φ is an isomorphism.

(v) By Exercise 8.13, ω^k is again a primitive nth root of unity if and only if $\gcd(n,k) = 1$, and there are precisely $\varphi(n)$ choices for such $k \in \{0,\ldots,n-1\}$.

8.16 (i) follows from Exercise 8.15.

(ii) Let $a \in \mathbb{F}_q^\times$ be such that $a^{(q-1)/p_j} \neq 1$. Such an element exists since the polynomial $x^{(q-1)/p_j} - 1$ has at most $(q-1)/p_j < q-1$ roots in $\mathbb{F}_q^\times$. We let $b_j = a^{(q-1)/p_j^{e_j}}$. Then $b_j^{p_j^{e_j}} = a^{q-1} = 1$, by Fermat's little theorem, and $b_j^{p_j^{e_j-1}} = a^{(q-1)/p_j} \neq 1$, and (i) yields the claim. (j was called i in the 1999 edition.)

(iii) Let $m = \mathrm{ord}(a)\,\mathrm{ord}(b)$. Since $(ab)^m = (a^{\mathrm{ord}(a)})^{\mathrm{ord}(b)}(b^{\mathrm{ord}(b)})^{\mathrm{ord}\,a} = 1$, we have that $\mathrm{ord}(ab)$ divides m. Suppose that there is a prime divisor p of m, say of $\mathrm{ord}(a)$, with $(ab)^{m/p} = 1$. Now $a^{\mathrm{ord}(b)}$ is an element of order $\mathrm{ord}(a)$, by Exercise 8.13 (iii), but

$$(a^{\mathrm{ord}(b)})^{\mathrm{ord}(a)/p} = a^{m/p} \cdot (b^{\mathrm{ord}(b)})^{\mathrm{ord}(a)/p} = (ab)^{m/p} = 1.$$

Therefore $m = \mathrm{ord}(ab)$.

(iv) is immediate from (ii) and (iii).

(v) follows from (iv) with $n = q-1$. (n was called t in the 1999 edition.)

8.18 In a field F such that n is a unit in F, the notions "primitive nth root of unity" and "element of multiplicative order n" coincide (Exercise 8.15). Now the order of $\mathbb{F}_q^\times$ is $q-1$, and by Lagrange's theorem, the order of any element in $\mathbb{F}_q^\times$ divides $q-1$. Exercise 8.16 shows that the condition $n \mid q-1$ is also sufficient for an element of order n to exist in $\mathbb{F}_q^\times$.

If n is not a unit in $\mathbb{F}_q$, then the characteristic p of $\mathbb{F}_q$ divides n, say $n = pm$ for some $m \in \mathbb{N}$. We let $\omega \in \mathbb{F}_q$ be any nth root of unity. Then $(\omega^m - 1)^p = \omega^n - 1 = 0$, so that $\omega^m - 1$ is a zero divisor and ω is not a primitive nth root of unity.

8.19 (i) Let $\omega_p, \omega_q \in \mathbb{Z}$ be primitive kth and lth roots of unity modulo p, q, respectively. If we compute $\omega \in \mathbb{Z}$ such that $\omega \equiv \omega_p \bmod p$ and $\omega \equiv \omega_q \bmod q$, then ω is a primitive mth root of unity modulo pq.

(ii) By the Chinese Remainder Theorem, an element $\omega \in \mathbb{Z}$ is a primitive kth root of unity modulo pq if and only if it is a primitive kth root of unit modulo p and modulo q. (It is not sufficient that ω be a primitive kth root of unity modulo p and only a kth root of unity modulo q, say of order k/t for some prime t dividing k, since then $\omega^{k/t} - 1$ is a zero divisor modulo pq.) By Lemma 8.8, the existence of primitive kth roots of unity modulo p and modulo q is equivalent to $k \mid (p-1)$ and $k \mid (q-1)$, and the claim follows.

8.20 (i) Let $q \in \mathbb{N}$ be such that $n = qm + r$ and $0 \le r < m$, and x an indeterminate. We have $x^m \equiv 1 \bmod x^m - 1$, and hence $x^n - 1 = (x^m)^q x^r - 1 \equiv x^r - 1 \bmod x^m - 1$. Now $\deg(x^r - 1) < \deg(x^m - 1)$, and hence $x^r - 1 = x^n - 1 \text{ rem } x^m - 1$. Since $x^r - 1 = 0$ if and only if $r = 0$, we have in particular that $x^m - 1$ divides $x^n - 1$ if and only if $m \mid n$. Now let $r_0 = n, r_1 = m > r_2 > \ldots > r_\ell > r_{\ell+1} = 0$ be the remainders in the Euclidean Algorithm for n and m. Then

$$\deg(x^m - 1) > \deg(x^{r_2} - 1) > \cdots > \deg(x^{r_\ell} - 1) > \deg(x^{r_{\ell+1}} - 1) = \infty,$$

and hence $x^n - 1, x^m - 1, \ldots, x^{r_\ell} - 1$ are the remainders in the Euclidean Algorithm for $x^n - 1$ and $x^m - 1$, and the claim follows. The proof for an integer $a \ge 2$ follows from this by substituting $x = a$ in the Euclidean Algorithm for $x^n - 1$ and $x^m - 1$ and noting that $r < m \iff \deg(x^r - 1) < \deg(x^m - 1) \iff a^r - 1 < a^m - 1$ and $r = 0 \iff x^m - 1 = 0 \iff a^m - 1 = 0$.

(ii) We have $2^n \equiv 1 \bmod M_n$, whence 2 is an nth root of unity modulo M_n. We first let n be prime. By Fermat's little theorem 4.9, $2^{n-1} \equiv 1 \bmod n$, and hence $M_n = 2 \cdot 2^{n-1} - 1 \equiv 1 \bmod n$. Thus $\gcd(M_n, n) = 1$, and n is a unit modulo M_n. Moreover, n is its only prime divisor and $2^{n/n} - 1 = 1$ is not a zero divisor modulo M_n, so that 2 is a primitive nth root of unity. Conversely, if n has a proper prime divisor t, then $\gcd(2^n - 1, 2^{n/t} - 1) = 2^{\gcd(n,n/t)} - 1 = 2^{n/t} - 1$, by (i). Now Exercise 4.14 implies that $2^{n/t} - 1$ is a zero divisor modulo M_n if $t < n$, and 2 is not a primitive nth root of unity modulo M_n.

8.24 (i) Let $T(n)$ denote the cost. Step 2 takes $2n$ additions and subtractions, step 3 costs $2T(n/2)$ ring operations plus $3n/2$ multiplications by powers of ω, and step 4 takes another n additions and n divisions by 2. Thus $T(n) = 2T(n/2) + 11n/2$ if $n > 1$, and together with $T(1) = 0$, the claim follows from Lemma 8.2.

(ii) Using that classical multiplication of two polynomials of degree less than 2^d costs $2 \cdot 2^{2d} - 2 \cdot 2^d + 1$ operations, we obtain $\gamma(d) = 2 \cdot 2^d - (1 + 11d/2) + 2^{-d}$. Taking derivatives with respect to d yields the minimal value when

$$2^d = (11/2 + (121/4 + 8(\ln 2)^2)^{1/2})/4\ln 2 \approx 4.08967,$$

and of the two nearest integers 2 and 3, $d = 2$ yields the smaller value $\gamma(2) = -15/4$. The cost of the hybrid algorithm is then $\frac{11}{2} n(\log n - 15/4)$. As an example, for $n \leq 128$ this is at most half of the cost from (i).

8.25 ALGORITHM 8.31 Fast Fourier Transform (FFT).
Input: $n = 2^k \in \mathbb{N}_{>0}$ with $k \in \mathbb{N}$, $f = \sum_{0 \leq j < n} f_j x^j \in R[x]$, and the powers $\omega, \omega^2, \ldots, \omega^{n-1}$ of a primitive nth root of unity $\omega \in R$.
Output: $\mathrm{DFT}_\omega(f) = (f(1), f(\omega), \ldots, f(\omega^{n-1})) \in R^n$.

1. **if** $n = 1$ **then return** (f_0)
2. write $f = a(x^2) + x \cdot b(x^2)$ with $a, b \in R[x]$ of degree less than $n/2$
3. **call** the algorithm recursively to compute

$$(\alpha_j)_{0 \leq j < n/2} = \mathrm{FFT}\Big(\frac{n}{2}, a, \omega^2, \omega^4, \ldots, \omega^n\Big)$$
$$(\beta_j)_{0 \leq j < n/2} = \mathrm{FFT}\Big(\frac{n}{2}, b, \omega^2, \omega^4, \ldots, \omega^n\Big)$$

4. **for** $j = 0, \ldots, (n/2) - 1$ **do** $\gamma_j \longleftarrow \alpha_j + \omega^j \beta_j, \quad \gamma_{j+n/2} \longleftarrow \alpha_j - \omega^j \beta_j$
5. **return** $(\gamma_0, \ldots, \gamma_{n-1})$

8.26 (i) ALGORITHM 8.32 Three-adic FFT.
Input: $k \in \mathbb{N}$, $n = 3^k$, $f = \sum_{0 \leq j < n} f_j x^j \in R[x]$, and the powers $\omega, \omega^2, \ldots, \omega^{n-1}$ of a primitive nth root of unity $\omega \in R$.
Output: $\mathrm{DFT}_\omega(f) = (f(1), f(\omega), f(\omega^2), \ldots, f(\omega^{n-1})) \in R^n$.

1. **if** $k = 0$ **then return** f
2. $\xi \longleftarrow \omega^{n/3}$
   $r_0 \longleftarrow \sum_{0 \le j < n/3} (f_j + f_{j+n/3} + f_{j+2n/3}) x^j$
   $r_1 \longleftarrow \sum_{0 \le j < n/3} (f_j + f_{j+n/3}\xi + f_{j+2n/3}\xi^2) \omega^j x^j$
   $r_2 \longleftarrow \sum_{0 \le j < n/3} (f_j + f_{j+n/3}\xi^2 + f_{j+2n/3}\xi^4) \omega^{2j} x^j$
3. **call** the algorithm recursively to evaluate r_0, r_1, r_2 at the powers of ω^3
4. **return**

$$\Big(r_0(1), r_1(1), r_2(1), r_0(\omega^3), r_1(\omega^3), r_2(\omega^3), \ldots, r_0(\omega^{n-3}), r_1(\omega^{n-3}), r_2(\omega^{n-3})\Big)$$

Correctness is clear if $k = 0$. If $k \ge 1$, we have to show that $f(\omega^{3\ell}) = r_0(\omega^{3\ell})$, $f(\omega^{3\ell+1}) = r_1(\omega^{3\ell})$, and $f(\omega^{3\ell+2}) = r_2(\omega^{3\ell})$ for $0 \le \ell < n/3$. For example, the last assertion follows from $\xi^3 = \omega^n = 1$ and

$$\begin{aligned} f(\omega^{3\ell+2}) &= \sum_{0 \le j < n/3} f_j \omega^{(3\ell+2)j} + \sum_{n/3 \le j < 2n/3} f_j \omega^{(3\ell+2)j} + \sum_{2n/3 \le j < n} f_j \omega^{(3\ell+2)j} \\ &= \sum_{0 \le j < n/3} (f_j \omega^{(3\ell+2)j} + f_{j+n/3} \omega^{(3\ell+2)j} \xi^{(3\ell+2)} + f_{j+2n/3} \omega^{(3\ell+2)j} \xi^{2(3\ell+2)}) \\ &= \sum_{0 \le j < n/3} (f_j + f_{j+n/3}\xi^2 + f_{j+2n/3}\xi^4) \omega^{2j} \omega^{3\ell j} = r_2(\omega^{3\ell}). \end{aligned}$$

(iii) The cost for computing the coefficients of r_0, r_1, r_2 in step 2 is $2n$ multiplications by powers of ω and $2n$ additions. Thus $T(1) = 0$ and $T(n) = 3T(n/3) + 4n$ if $n > 1$, whence $T(n) = 4n \log_3 n$.

8.28 $\omega - 1$ is a zero divisor.

8.29 (i) Induction on p reveals that $f_p = q \cdot (x-1) + p$, where $q = x^{p-2} + 2x^{p-3} + \cdots + (p-2)x + (p-1)$. Thus $q \cdot (x-1) \equiv -p \bmod f_p$. If a is the inverse of p in R, then $-aq$ is the inverse of $x-1$ modulo f_p. Similarly, if $a \in R \setminus \{0\}$ is such that $ap = 0$ in R, then $aq \not\equiv 0 \bmod f_p$ in $R[x]$ since q is monic and of smaller degree than f_p, and $aq \cdot (x-1) \equiv 0 \bmod f_p$.

(iii) We have $\omega^{pn} - 1 = \Phi_{pn}(\omega) \cdot (\omega^n - 1) = 0$, and pn is a unit in R since p is, with inverse $p^{-(k+1)}$. Now

$$(\omega^n - 1)(\omega^{(p-2)n} + 2\omega^{(p-3)n} + \cdots + (p-2)\omega^n + (p-1)) = f_p(\omega^n) - p = -p,$$

by (i) and since $f_p(\omega^n) = \Phi_{pn}(\omega) = 0$. Since $-p$ is a unit, so is $\omega^n - 1$, and the claim follows since p is the only prime divisor of pn.

8.30 In the 1999 edition, the text of the exercise contains several typos, and we first give a corrected version of it.

In this exercise, we discuss Schönhage's (1977) 3-adic variant of Algorithm 8.20. It works over any (commutative) ring R such that 3 is a unit in R, so in particular over a field of characteristic 2.

ALGORITHM 8.30 Schönhage's algorithm.
Input: Two polynomials $f,g \in R[x]$ of degree less than $2n = 2\cdot 3^k$ for some $k \in \mathbb{N}$, where R is a (commutative) ring and 3 is a unit in R.
Output: $h \in R[x]$ such that $fg \equiv h \bmod (x^{2n}+x^n+1)$ and $\deg h < 2n$.

1. **if** $k \le 2$ **then**
   call the classical algorithm 2.3 (or Karatsuba's algorithm 8.1) to compute $f\cdot g$
   return fg rem $x^{2n}+x^n+1$

2. $m \longleftarrow 3^{\lceil k/2\rceil}, \quad t \longleftarrow n/m$
   let $f',g' \in R[x,y]$ with $\deg_x f', \deg_x g' < m$ such that $f = f'(x,x^m)$ and $g = g'(x,x^m)$

3. let $D = R[x]/\langle x^{2m}+x^m+1\rangle$
   if $m = t$ **then** $\eta \longleftarrow x \bmod (x^{2m}+x^m+1)$ **else** $\eta \longleftarrow x^3 \bmod (x^{2m}+x^m+1)$
   { η is a primitive $3t$th root of unity }
   $f^* \longleftarrow f' \bmod (x^{2m}+x^m+1), \quad g^* \longleftarrow g' \bmod (x^{2m}+x^m+1)$

4. **for** $j = 1,2$ **do**
   $f_j \longleftarrow f^*$ rem $y^t - \eta^{jt}, \quad g_j \longleftarrow g^*$ rem $y^t - \eta^{jt}$
   call the fast convolution algorithm 8.16 with $\omega = \eta^3$ to compute $h_j \in D[y]$ of degrees less than t such that

$$f_j(\eta^j y)g_j(\eta^j y) \equiv h_j(\eta^j y) \bmod y^t - 1$$

   { the DFTs are performed by the 3-adic FFT algorithm from Exercise 8.26, and Algorithm 8.30 is used recursively for multiplications in D }

5. $h^* \longleftarrow \dfrac{1}{3}(y^t(h_2-h_1)+\eta^{2t}h_1-\eta^t h_2)(2\eta^t+1)$
   let $h' \in R[x,y]$ with $\deg_x h' < 2m$ such that $h^* = h' \bmod (x^{2m}+x^m+1)$
   $h \longleftarrow h'(x,x^m)$ rem $(x^{2n}+x^n+1)$
   return h

(i) Use Exercise 8.29 to prove that the algorithm works correctly.

(ii) Let $T(k)$ denote the cost of the algorithm for $n = 3^k$. Prove that $T(k) \le 2\cdot 3^{\lfloor k/2\rfloor}T(\lceil k/2\rceil) + (c+48(\lfloor k/2\rfloor+1/2))3^k$ for $k > 2$ and some constant $c \in \mathbb{N}$, and conclude that $T(k)$ is at most $24\cdot 3^k\cdot k\cdot\log k + O(3^k\cdot k) = 24n\log_3 n\log_2\log_3 n +$

$O(n\log n)$. Hint: Consider the function $S(k) = (3^{-k}T(k)+c)/(k-1)$, and prove that $S(k) \le S(\lceil k/2\rceil)+24$ if $k > 2$.

Solution:

(i) We proceed by induction on k. There is nothing to prove if $k \le 2$, and we assume that $k \ge 3$. Exercise 8.29 shows that $x \bmod x^{2m}+x^m+1 \in D$ is a primitive $3m$th root of unity in D, and hence η is a primitive $3t$th root of unity. By induction, the results of the recursive calls in step 3 are correct. Substituting $\eta^{-j}y$ for y, we obtain

$$f^*g^* \equiv f_1g_1 \equiv h_1 \bmod y^t-\eta^t, \quad f^*g^* \equiv f_2g_2 \equiv h_2 \bmod y^t-\eta^{2t},$$

since $(\eta^{-1}y)^t-1 = \eta^{-t}(y^t-\eta^t)$ and $(\eta^{-2}y)^t-1 = \eta^{-2t}(y^t-\eta^{2t})$. Using the fact that $\eta^{2t}+\eta^t+1 = 0$, a calculation shows that

$$h^* \equiv (\eta^t(h_2-h_1)+\eta^{2t}h_1-\eta^th_2)\frac{2\eta^t+1}{3} = h_1 \equiv f^*g^* \bmod (y^t-\eta^t),$$

and similarly $h^* \equiv f^*g^* \bmod y^t-\eta^{2t}$. Now $(y^t-\eta^t)(y^t-\eta^{2t}) = y^{2t}+y^t+1$ and $\gcd(y^t-\eta^{2t},y^t-\eta^t) = 1$ since $\eta^{2t}-\eta^t$ is a unit, and the Chinese Remainder Theorem implies that $h^* \equiv f^*g^* \bmod y^{2t}+y^t+1$. Now

$$\begin{aligned} h' \bmod x^{2m}+x^m+1 = h^* &= f^*g^* \operatorname{rem} y^{2t}+y^t+1 \\ &= (f'g' \operatorname{rem} y^{2t}+y^t+1) \bmod x^{2m}+x^m+1, \end{aligned}$$

and since the coefficients of f' and g' have degrees less than m in x, the coefficients of $f'g' \operatorname{rem} y^{2t}+y^t+1$ have degree less than $2m$ in x, and $\deg_x h' < 2m$ implies that $h' = f'g' \operatorname{rem} y^{2t}+y^t+1$. Finally, plugging in x^m for y (or equivalently, computing modulo $y-x^m$), we have

$$h \equiv h'(x,x^m) = f'(x,x^m)g'(x,x^m) \operatorname{rem} x^{2n}+x^n+1 \equiv fg \bmod x^{2n}+x^n+1.$$

(ii) The cost for step 1 is $O(1)$, and steps 2 and 3 are for free. By Exercise 8.26, the two convolutions modulo y^t-1 in step 4 cost $6\cdot 4t\log_3 t$ additions and multiplications by powers of η, $2t$ divisions by t, and $2t$ "essential" multiplications in D, each of the latter taking $T(\lceil k/2\rceil)$ operations in R. The reductions modulo $y^t-\eta^{jt}$ for $j = 1,2$ amount to $4t$ multiplications by powers of η and the same number of additions in D, and computing $f_j(\eta^jy), g_j(\eta^jy)$ from f_j, g_j and h_j from $h_j(\eta^jy)$ for $j = 1,2$ takes another $6t$ multiplications by powers of η. Finally, in step 5 we have $4t$ multiplications by powers of η, $4t$ additions, and $4t$ multiplications by 2 or $1/3$ in D for the computation of h^*, plus at most $2mt$ additions in R to compute h from h'.

The cost for one addition in D is $2m$ additions in R. Since $x^{2m}+x^m+1$ divides $x^{3m}-1$, one multiplication of $a \bmod x^{2m}+x^m+1 \in D$, with $a \in R[x]$ of degree less

than $2m$, by a power η^j can be done by first computing $ax^j \bmod x^{3m}-1$ or $ax^{3j} \bmod x^{3m}-1$, respectively, which is just a cyclic shift of coordinates and hence for free, and one subsequent reduction modulo $x^{2m}+x^m+1$, taking $2m$ additions in R. One division by t or one multiplication by $2/3$ in D amounts to $2m$ divisions or multiplications in R, respectively. Putting it all together, we have

$$\begin{aligned} T(k) &\le 2t \cdot T\Big(\Big\lceil \frac{k}{2} \Big\rceil\Big) + (48\log_3 t + 58)mt \\ &\le 2 \cdot 3^{\lfloor k/2 \rfloor} T\Big(\Big\lceil \frac{k}{2} \Big\rceil\Big) + \Big(82 + 48\Big(\Big\lfloor \frac{k}{2} \Big\rfloor - \frac{1}{2}\Big)\Big) 3^k \end{aligned}$$

if $k > 2$, that is, $c = 82$. Letting $S(k) = (3^{-k}T(k)+c)/(k-1)$, we obtain

$$S(k) \le S\Big(\Big\lceil \frac{k}{2} \Big\rceil\Big) + 24 \le \cdots \le S(2) + 24(\lceil \log k \rceil - 1)$$

for $k > 2$, by induction, and hence

$$\begin{aligned} T(k) &= 3^k((k-1)S(k) - c) \\ &\le 24 \cdot 3^k (k-1)(\lceil \log k \rceil - 1) + S(2) \cdot 3^k (k-1) - c\,3^k \\ &\in 24 \cdot 3^k k \log k + O(3^k k) = 24n \log_3 n \log_2 \log_3 n + O(n \log n). \end{aligned}$$

8.31 (i) Let $n = p^{k+1}$ for a prime $p \in \mathbb{N}$ and some $k \in \mathbb{N}$, $f_p = x^{p-1} + \cdots + x + 1 \in R[x]$, $\Phi_n = f_p(x^{p^k}) \in R[x]$, $\omega = x \bmod \Phi_n \in R[x]/\langle \Phi_n \rangle$, as in Exercise 8.29, and $1 \le l = p^m t < n$, with $0 \le m \le k$ and $p \nmid t$. The p-adic expansion of all exponents j yields the formula

$$\sum_{0 \le j < n} x^j = f_p(x)\, f_p(x^p) \cdots f_p(x^{p^k})$$

in $R[x]$. (The reader familiar with cyclotomic polynomials will recognize this as a special case of Lemma 14.46.) If we plug in $x = \omega^l$, we find the factorization

$$\sum_{0 \le j < n} \omega^{jl} = f_p(\omega^l)\, f_p(\omega^{pl}) \cdots f_p(\omega^{p^k l}) \tag{10}$$

in R. We claim that $f_p(\omega^{p^{k-m}l}) = f_p(\omega^{nt/p}) = 0$. For each $j \in \{1, \ldots, p-1\}$ there is a unique $j^* \in \{1, \ldots, p-1\}$ such that $tj \equiv j^* \bmod p$. Since $\Phi_n \mid (x^n - 1)$, we have $\omega^n = 1$, and

$$f_p(\omega^{nt/p}) = 1 + \sum_{1 \le j < p} \omega^{ntj/p} = 1 + \sum_{1 \le j^* < p} \omega^{nj^*/p} = f_p(\omega^{n/p}) = \Phi_n(\omega) = 0,$$

and the claim is proved. Together with (10), this shows that Lemma 8.7 (ii) is true for the above value of ω, and hence also the second conclusion of Theorem 8.13. (j and t were called i and j, respectively, in the 1999 edition.)

(ii) The recursive calls in step 3 of the modified algorithm return $2^{e(\lfloor k/2\rfloor+1)}$ times the result of the original algorithm. The modified convolution algorithm multiplies this factor by $t = 2^{\lceil k/2\rceil}$. Noting that $\lfloor (k+1)/2\rfloor = \lceil k/2\rceil$ and $\lceil (k+1)/2\rceil = \lfloor k/2\rfloor + 1$, we obtain

$$e(k+1) = e\Big(\Big\lfloor \frac{k+1}{2}\Big\rfloor + 1\Big) + \Big\lceil \frac{k+1}{2}\Big\rceil = e\Big(\Big\lceil \frac{k}{2}\Big\rceil + 1\Big) + \Big\lfloor \frac{k}{2}\Big\rfloor + 1.$$

Letting $S(k) = e(k+1) - k$, we have

$$S(k) = S\Big(\Big\lceil \frac{k}{2}\Big\rceil\Big) + 1 = \cdots = S(1) + \lceil \log k\rceil$$

for $k \geq 1$, by induction, and the claim follows from $S(1) = e(2) - 1 = -1$.

8.32 We have $2^n \equiv -1 \bmod 2^n+1$ and $2^{2n} \equiv (-1)^2 = 1 \bmod 2^n+1$, and 2 is a $2n$th root of unity. Assume first that $n = 2^k$ for some $k \in \mathbb{N}$. Since 2^n+1 is odd, n is a unit modulo 2^n+1. Moreover, $2^n - 1 \equiv -2 \bmod 2^n+1$ is a unit, and therefore 2 is a primitive $2n$th root of unity, since 2 is the only prime divisor of $2n$.

Conversely, if $n = pm$ for an odd prime p and $m \in \mathbb{N}_{\geq 1}$, then substituting 2^m for x in the equation

$$(x+1)\cdot(x^{p-1} - x^{p-2} + \cdots - x + 1) = x^p + 1$$

in $\mathbb{Z}[x]$ yields $(2^m+1)\cdot q = 2^n+1$, where

$$2^n > q = 2^{m(p-1)} - 2^{m(p-2)} + \cdots - 2^m + 1 > 0.$$

Thus

$$(2^{2m}-1)q = (2^m-1)(2^n+1)q \equiv 0 \bmod 2^n+1,$$

and $2^{2m}-1$ is a zero divisor modulo 2^n+1. Finally, Lemma 8.7 implies that 2 is not a primitive $2n$th root of unity modulo 2^n+1.

8.33 We have $\mathsf{M}(mn)/mn \geq \mathsf{M}(n)/n$, and the first claim follows from multiplying up by mn. Similarly, $\mathsf{M}(m+n)/(m+n) \geq \mathsf{M}(m)/m$ implies

$$\mathsf{M}(m) \leq m\frac{\mathsf{M}(m+n)}{m+n} \text{ and } \mathsf{M}(n) \leq n\frac{\mathsf{M}(m+n)}{m+n},$$

and the second claim follows by summing the two inequalities. Finally, $\mathsf{M}(n)/n \geq \mathsf{M}(1)/1 = 1$, and the last claim follows.

8.34 Let R be a ring. To multiply $a, b \in R[x]$ of degrees at most n, we write them as $a = a_n x^n + a^*$ and $b = b_n x^n + b^*$, with $a_n, b_n \in R$ and $a^*, b^* \in R[x]$ of degrees less than n. Then we compute $ab = a_n b_n x^{2n} + a_n x^n b^* + b_n x^n a^* + a^* b^*$. This takes $2n+1$ multiplications for the first three summands, $\mathsf{M}(n)$ ring operations for the last product, and $2n-1$ additions for adding everything together.

8.35 (i) Let R be a ring and $a, b \in R[x]$ with $\deg a < n$ and $\deg b < kn$. We divide b into k blocks of degree less than n each: $n = \sum_{0 \le j < k} b_j x^{nj}$, with $b_j \in R[x]$ and $\deg b_j < n$ for all j. Then we compute $ab = \sum_{0 \le j < k} ab_j x^{nj}$. This takes $k \cdot \mathsf{M}(n)$ ring operations for the k products ab_j, plus $(k-1)(n-1)$ additions in R for summing up the overlapping blocks $ab_j x^{nj}$.

8.36 (i) Let $f = \sum_{0 \le j < n} f_j x^j$ and $g = \sum_{0 \le j < n} g_j x^j$ in $\mathbb{Z}[x]$, with all $f_j, g_j \in \mathbb{Z}$. Then the absolute value of the mth coefficient of fg is

$$\left| \sum_{0 \le j, m-j < n} f_j g_{m-j} \right| \le n \cdot 2^l \cdot 2^l = 2^{k+2l} \le 2^{n-1}.$$

(ii) Let $f^* = f \bmod 2^n + 1$ and $g^* = g \bmod 2^n + 1$ in $R[x]$. By Exercise 8.32, the fast convolution algorithm 8.16 with $\omega = 2$ computes $h^* \in R[x]$ of degree less than n such that $h^* \equiv f^* g^* \bmod x^n + 1$, and $\deg(fg) < n$ implies that $h^* = f^* g^* = fg \bmod 2^n + 1$. By (i), the coefficients of fg are at most 2^{n-1} in absolute value and can be uniquely recovered from those of h^*.

The cost for this is $O(n \log n)$ additions and multiplications by powers of ω plus $O(n)$ essential multiplications in R and the same number of divisions by n, by Theorem 8.18. A multiplication by a power of ω corresponds to a cyclic shift in the binary representation with a sign inversion of the wrapped around coordinates, and hence one addition or one multiplication by a power of ω takes $O(n)$ word operations. The same is true for a division by n. One multiplication in R can be done with $O(n \log n \operatorname{loglog} n)$ word operations, by Theorem 8.24 (one reduction of a modulo $2^n + 1$ corresponds to subtracting the upper part of a from the lower part in the binary representation), and the claim follows.

Chapter 9

9.1 $f^{-1} \equiv 1 + 2x + 3x^2 + 4x^3 + 5x^4 + 6x^5 + 7x^6 + 8x^7 \bmod x^8$.

9.2 $94^{-1} \equiv 349 \bmod 3^8$.

9.6 The cost for step $i = r - j$ is $\mathsf{M}(\lceil l 2^{-j} \rceil) + \mathsf{M}(\lceil l 2^{-j-1} \rceil) + \lfloor l 2^{-j-1} \rfloor$ operations or $\frac{3}{2}\mathsf{M}(\lfloor l 2^{-j} \rfloor) + O(l 2^{-j})$. Ignoring linear terms, which contribute only $O(l)$ to the total, we obtain

$$\frac{3}{2} \sum_{0 \le j < r} \mathsf{M}(\lfloor l 2^{-j} \rfloor) \le \frac{3}{2} \mathsf{M}(l) \sum_{0 \le j < r} 2^{-j} \le 3\mathsf{M}(l).$$

9.7 Let $d = \deg p$ for short.

(i) Computing $p^{2^{i+1}}$ from p^{2^i} takes at most $\mathsf{M}(2^i d) + 4 \cdot 2^i d \le 2^{i-r}(\mathsf{M}(n) + 4n)$ operations in D, by Exercise 8.34. Summing this for $0 \le i < r$ yields the claim.

(ii) We use Algorithm 9.3 to compute $\mathrm{rev}(p)^{-1}$ rem x^d, taking $3\mathsf{M}(d)+O(d)$ or $3\cdot 2^{-r}(\mathsf{M}(n)+O(n))$ operations in D, by Exercise 9.6. Then we use Exercise 9.5 (i) to compute $\mathrm{rev}(p^{2^{i+1}})^{-1}$ mod $x^{2^{i+1}d}$ from $\mathrm{rev}(p^{2^i})^{-1}$ mod $x^{2^i d}$, for $0\le i<r$. This takes $2\mathsf{M}(2^i d)+\mathsf{M}(2^{i+1}d)+O(2^i d)$ or $4\cdot 2^{i-r}(\mathsf{M}(n)+O(n))$ operations in D for each value of i, in total at most $4\mathsf{M}(n)-4\cdot 2^{-r}\mathsf{M}(n)+O(n)$, and the claim follows.
(iii) Using the precomputed data, computing f rem p^{2^i} from f rem $p^{2^{i+1}}$ takes $2\mathsf{M}(2^i d)+O(2^i d)$ or $2\cdot 2^{i-r}(\mathsf{M}(n)+O(n))$ operations in D, and the claim follows by summing up.

9.8 (ii) Let $R=D[x]$ and $d=\deg p$. We first perform the precomputations as in Exercise 9.7, at a cost of $7\mathsf{M}(ld)+O(ld)$ operations in D. The cost for the ith iteration of step 2 is $\mathsf{M}(2^{i-1}d)$ for squaring g_{i-1}, $\mathsf{M}(2^i d)$ for multiplying the result by f rem p^{2^i}, $O(2^i d)$ for subtracting this from $2g_{i-1}$, and $2\mathsf{M}(2^i d)+O(2^i d)$ for the final reduction modulo p^{2^i}. Together, this amounts to at most $\frac{7}{2}\mathsf{M}(2^i d)+O(2^i d)$ or $\frac{7}{2}2^{i-r}(\mathsf{M}(ld)+O(ld))$ operations in D. Summing this for $1\le i\le r$ yields at most $7\mathsf{M}(ld)+O(ld)$.

9.9 Let $f=\sum_{i\ge 0} f_i x^i$ and $h=\sum_{i\ge 0} h_i x^i$ the power series inverse of f. Then $g_i = h$ rem x^i and $h_{i+1}=-f_0^{-1}(f_1 h_i+f_2 h_{i-1}+\cdots+f_{i+1}h_0)$.

9.10 The Newton formula is then $g_i\equiv fg_{i-1}^2$ mod x^{2^i}. Squaring is for free, and hence the ith step costs $\mathsf{M}(2^i)$. The total cost is then $\sum_{1\le i\le r}\mathsf{M}(2^i)\le 2\mathsf{M}(l)$.

9.11 $fh=fg\cdot(e^{d-1}+\cdots+e+1)=-(e-1)(e^{d-1}+\cdots+e+1)=-e^d+1\equiv 1$ mod x^{kd}. The cost is $O(\mathsf{M}(l))$.

9.15 (i) We have $g_i\equiv g_{i-1}$ mod $x^{2^{i-1}}$ in Algorithm 9.3, so that we need only consider the coefficients of the upper half $g_i^*=g_i$ quo $x^{2^{i-1}}$. Since $x^{2^{i-1}}\mid(1-fg_{i-1})$, we have $g_i^*\equiv-(fg_{i-1}$ quo $x^{2^{i-1}})g_{i-1}$ mod $x^{2^{i-1}}$, and hence

$$\|g_i^*\|_\infty\le 2^{i-1}\|fg_{i-1}\|_\infty\cdot\|g_{i-1}\|_\infty\le 2^{2(i-1)}\|f\|_\infty\cdot\|g_{i-1}\|_\infty^2<2^{2(i-1)+l}\|g_{i-1}\|_\infty^2.$$

(ii) Taking derivatives and multiplying by x, we find

$$\sum_{0\le j<i} jx^j=x\frac{-ix^{i-1}(1-x)+(1-x^i)}{(1-x)^2},$$

and plugging in $x=1/2$ yields

$$\sum_{0\le j<i} j2^{-j}=2(-i2^{-i}+1-2^{-i})=2-(i+1)2^{1-i}\le 2.$$

(iii) We have $S(0)=\log|g_0|=0$. From (i), we find inductively

$$\begin{aligned}S(i)&\le 2(i-1)+l+2S(i-1)\le 2(i-1)+l+2(2(i-2)+l+2S(i-2))\\&=2^1(i-1)+2^2(i-2)+(2^2-1)l+2^2S(i-2)\le\cdots\\&\le 2^i\sum_{0\le j<i} j2^{-j}+(2^i-1)l+2^iS(0)\le(2+l)2^i,\end{aligned}$$

by (ii).

(iv) Let $\deg_y a, \deg_y b \le l$. Then we have

$$\deg_y g_i \le l + 2\deg_y g_{i-1} \le \cdots \le (2^i - 1)l + 2^i \deg_y g_0 \le 2^i l.$$

9.17 The (x^2+1)-adic representation is $(x, -7x, 21x, -35x, 35x, -21x, 7x, -x)$.

9.18 If we denote the hexadecimal digits $10, 11, \ldots, 15$ by $\mathrm{A}, \mathrm{B}, \ldots, \mathrm{F}$, then the hexadecimal representation of 64180 is FAB4.

9.20 We first precompute $p^2, p^4, \ldots, p^{k/2}$, at a cost of $\mathsf{M}(km/2) + O(km)$ ring operations, by Exercise 9.7 (i). To compute the coefficients of a, we recursively compute the coefficients of a quo $p^{k/2}$ and a rem $p^{k/2}$, and then calculate $a = (a \text{ quo } p^{k/2}) \cdot p^{k/2} + (a \text{ rem } p^{k/2})$. Denote the cost for this by $T(k)$. Then $T(1) = 0$ and $T(k) = 2T(k/2) + \mathsf{M}(km/2) + km/2$ if $k > 1$, which evaluates to $T(k) = (\mathsf{M}(km/2) + km/2)\log k$. The claim follows from $\mathsf{M}(km/2) \le \mathsf{M}(km)/2$.

9.27 (i) follows by induction on n, (iii) by induction on r, (ii) is a special case of (iii), and (iv) follows from (ii) by dividing by $f_1 \cdots f_r$.

9.29 If the denominator of φ is not divisible by $y - g$, then the Taylor expansion around g, as in Lemma 9.20, exists with a rational function $\psi \in R(y)$ whose denominator is not divisible by $y - g$. Then Lemma 9.21 holds, and the Newton iteration algorithm 9.22 works if φ and φ' are defined at g_0 modulo p, $\varphi(g_0) \equiv 0 \bmod p$, and $\varphi'(g_0)$ is invertible modulo p.

For $\varphi = fy - 1$, the Newton formula is $g_i = g_{i-1} - (fg_{i-1} - 1)/f = 1/f$, and it does not lead to an algorithm for computing $1/f$.

9.30 The roots of φ modulo 5 are 2 and 3, and the only root of φ in $\mathbb{Z}$ is 18.

9.31 By Exercise 9.7, we can precompute the powers $p^2, p^4, \ldots, p^{2^{r-1}}$ at a cost of $O(\mathsf{M}(l \deg p))$ operations in D. Then the cost for one multiplication of two polynomials in $D[x]$ modulo p^{2^i} is $O(\mathsf{M}(2^i \deg p))$ operations in D, by Corollary 9.7. The number of such modular multiplications and additions in the ith iteration of step 2 is $O(n)$, as in the proof of Theorem 9.25, and the claim follows from $\mathsf{M}(2^i \deg p) \le 2^{r-i}\mathsf{M}(2^r \deg p)$ by summing up.

9.36 $1 + 2x - 2x^2 + 3x^3 - 10x^4 + 28x^5 - 84x^6 + 264x^7$.

9.37 There is only one cube root of 2 modulo 5, namely 3, and 303 is the only cube root of 2 modulo 5^4.

9.39 We have $a^n = (a^{n/2})^2$ if n is even, and $a^n = (a^{(n-1)/2})^2 a$ if n is odd. Thus the cost for computing a^n is $T(\lfloor n/2 \rfloor)$ for the recursive call, plus $O(\mathsf{M}(nl))$ for one or two multiplications of numbers of length at most nl. The claim follows with $T(1) = 0$ by induction. In the polynomial case, the cost is $O(\mathsf{M}(n \deg a))$ ring operations.

9.40 (i) $\mathbb{Z}_p = \mathbb{F}_p$ is a field, and hence the polynomial $\varphi = y^2 - a$ has at most two roots in $\mathbb{F}_p$ and $S_p(a) \in \{0, 1, 2\}$. Each of the three cases occurs: If $p = 2$, then

$g = a$ is the only solution of $\varphi(g) \equiv 0 \bmod 2$, and $S_2(a) = 1$. If $p \mid a$, then $\varphi(g) \equiv 0 \bmod p \iff g \equiv 0 \bmod p$, and we have $S_p(a) = 1$ as well. In the remaining case where $2 \neq p \nmid a$, we have always $S_p(a) \neq 1$, for if g is a zero of φ modulo p, then so is $-g \not\equiv g \bmod p$. Each of the two cases occurs: for example, we have $S_3(2) = 0$ and $S_3(1) = 2$.

(ii) The claim is clear if $S_p(a) = 0$, so let us assume that $S_p(a) > 0$. If $2 \neq p \nmid a$ and $\varphi(g) \equiv 0 \bmod p$, then $g \not\equiv 0 \bmod p$, and hence $\varphi'(g) = 2g \not\equiv 0 \bmod p$. Algorithm 9.22 then shows that $S_p(a) \leq S_{p^e}(a)$, and the uniqueness of Newton iteration (Theorem 9.27) implies the reverse inequality.

If $2 \neq p \mid a$, then there is precisely one root of φ modulo p, by (i), but there may be none or more than one modulo p^e, as in the examples $a = p$ and $a = 0$, respectively, when $e > 1$.

(iii) If $\gcd(a,n) = 1$, then $p_i \nmid a$ for $1 \leq i \leq r$. Thus $S_n(a) = S_{p_1^{e_1}}(a) \cdots S_{p_r^{e_r}}(a) = S_{p_1}(a) \cdots S_{p_r}(a)$, by the Chinese Remainder Theorem and (ii). The last claim follows from $S_{p_i}(1) = 2$ for all i.

(iv) We have $50\,625 = 3^4 \cdot 5^4$ and

$$\begin{array}{rcl@{\qquad}rcl} 10\,001 & \equiv & 2 \bmod 3, & 10\,001 & \equiv & 1 \bmod 5, \\ 42\,814 & \equiv & 1 \bmod 3, & 42\,814 & \equiv & 4 \bmod 5, \\ 31\,027 & \equiv & 1 \bmod 3, & 31\,027 & \equiv & 2 \bmod 5, \\ 17\,329 & \equiv & 1 \bmod 3, & 17\,329 & \equiv & 4 \bmod 5. \end{array}$$

Since 2 has no square root modulo 3 and modulo 5, we conclude from (iii) that only 42 814 and 17 329 possess square roots modulo 50 625, and that they have exactly four such roots.

(v) We have $2025 = 3^4 5^2$, and the congruences $g^2 \equiv 91 \equiv 1 \bmod 3$ and $g^2 \equiv 91 \equiv 1 \bmod 5$ have solutions $g \equiv \pm 1 \bmod 3$ and $g \equiv \pm 1 \bmod 5$, respectively. Thus there are four distinct square roots of 91 modulo 2025, by (iii). Using 3-adic and 5-adic Newton iteration and the Chinese Remainder Algorithm, we obtain the four solutions $46, 521, 1504$, and 1979.

By (iii), there are precisely four square roots of 1 modulo 50 625. The two trivial ones are 1 and -1, and Newton iteration and the CRA yield the two other ones 8749 and -8749.

9.41 (iii) Similarly as in Exercise 9.40, we have $C_n(a) = C_{p_1}(a) \cdots C_{p_r}(a)$.

(iv) We have $225\,625 = 5^4 \cdot 19^2$. The number 1 is the only cube root modulo 5 of $11 \equiv 1 \bmod 5$, and there are three cube roots of 11 modulo 19, namely $5, 16, 17$. Thus there are precisely three cube roots of 11 modulo 225 625, by (i). Newton iteration and the Chinese Remainder Algorithm yield the three solutions $47771, 50271$, and 103396.

9.43 By extracting powers of 3 if necessary, we may assume that $3 \nmid a$. Let $\varphi = y^2 - a \in \mathbb{Z}[y]$. If $a \equiv 1 \bmod 3$, then $\varphi(1) \equiv 0 \bmod 3$ and $\varphi'(1) \equiv 2 \not\equiv \bmod 3$, and

$g_0 = 1$ is a starting solution. Otherwise, if $a \equiv 2 \bmod 3$, then a has no square root in $\mathbb{Z}$. In the first case, we call Algorithm 9.22 with $l = \lceil (\log_3 a)/2 \rceil$, so that $3^{2l} > a$, and it computes a square root $g \in \mathbb{N}$ of a modulo 3^l with $g < 3^l$, taking $O(\mathsf{M}(l))$ word operations, by Theorem 9.26. Finally, we check whether $g^2 = a$ or $(3^l - g)^2 = a$ in $\mathbb{Z}$, taking another $O(\mathsf{M}(l))$ word operations. If both tests fail, then the uniqueness of Newton iteration (Theorem 9.27) implies that a has no square root in $\mathbb{Z}$, as in Section 9.5.

The fourth root of 2313441 in $\mathbb{Z}$ is 39.

9.44 ALGORITHM 9.36 Perfect power testing.
Input: An integer $a \in \mathbb{N}_{>1}$.
Output: Integers $b, d, e, r \in \mathbb{N}$ such that $a = 2^d 3^e b^r$ and r is maximal with that property.

1. let $a = 2^d 3^e b$ with $\gcd(b, 6) = 1$, $\quad n \longleftarrow 2$, $\quad r \longleftarrow 1$
2. **while** $4^n < b$ **do**
3. $\quad \{\ 2^d 3^e b^r = a$ and b is not a kth power for $2 \le k < n\ \}$
   $\quad$ **call** the Newton iteration algorithm from Section 9.5 or Exercise 9.43 to check whether $b = c^n$ for some $c \in \mathbb{N}$
   $\quad$ **if** this is the case **then** $b \longleftarrow c$, $\quad r \longleftarrow rn$
   $\quad$ **else** $n \longleftarrow n + 1$
4. **return** b, d, e, r

The loop invariants follow by induction and imply the correctness. The cost for step 1 is $O(\log a)$ word operations, by Exercise 4.1, and the condition in step 2 can be checked with $O(1)$ word operations, by examining the length of b. One execution of step 3 takes $O(\mathsf{M}(\log a))$ word operations, and the number of iterations is at most $\log_4 a$. Then a is a perfect power if and only if $\gcd(d, e, r) > 1$.

9.47 The polynomial $\varphi'(y) - \varphi'(g) \in R[y]$ has g as a root and hence is divisible by $y - g$. Thus $\varphi'(h) - \varphi'(g) = q \cdot (h - g)$ for some $q \in R$, and $v(\varphi'(h) - \varphi'(g)) = v(q)v(h - g) \le \varepsilon < 1$. Then

$$v(\varphi'(h)) = v(\varphi'(h) - \varphi'(g) + \varphi'(g)) = \max\{v(\varphi'(h) - \varphi'(g)), v(\varphi'(g))\} = 1.$$

Chapter 10

10.3 Let $r = 2^k$. We have $\deg M_{i,j} = 2^i d$ for all i, j, and hence computing $M_{i,j}$ amounts to multiplying two polynomials of degree $2^{i-1}d$. This takes $\mathsf{M}(2^i d + 1) \le \mathsf{M}(2^{i-1}d) + 4 \cdot 2^{i-1} d$ ring operations, by Exercise 8.34. There are 2^{k-i} nodes at level i, so that the cost at level i is $2^{k-i}(\mathsf{M}(2^{i-1}d) + O(2^{i-1}d))$ or $\mathsf{M}(n/2) + O(n)$, and there are $k = \log r$ levels.

10.4 (i) We have $0 < p_i \le 1$, whence $-p_i \log p_i \ge 0$, for all i, and also the entropy is nonnegative. If $n > 1$, then $p_i < 1$ and $-p_i \log p_i > 0$ for all i, and the entropy is nonzero.

(ii) We have

$$\begin{aligned} H(p_1,\ldots,p_n) - \log n &= -\sum_{1\le i\le n} p_i \log p_i - \sum_{1\le i\le n} p_i \log n = \frac{1}{\ln 2} \sum_{1\le i\le n} p_i \ln \frac{1}{p_i n} \\ &\le \frac{1}{\ln 2} \sum_{1\le i\le n} p_i \Big(\frac{1}{p_i n} - 1\Big) = \frac{1}{\ln 2} \sum_{1\le i\le n} \Big(\frac{1}{n} - p_i\Big) = 0, \end{aligned}$$

with equality if and only if $p_i = 1/n$ for all i.

10.5 (i) Let t be a stochastic mobile. If $n = 1$ then the average depth is $0 = H(1) = H(p_1)$, and we assume that $n > 1$. Renumbering if necessary, we may assume that the leaves $n-1$ and n are children of the same node. Let t^* be the tree obtained from deleting those two leaves. Then t^* is again a stochastic mobile with leaf weights $p_1,\ldots,p_{n-2},p = p_{n-1}+p_n$, and its average depth d^* is inductively at least

$$\begin{aligned} H(p_1,\ldots,p_{n-2},p) &= -\sum_{1\le i\le n-2} p_i \log p_i - p\log p \\ &= H(p_1,\ldots,p_n) - p\log p + p_{n-1}\log p_{n-1} + p_n \log p_n \\ &= H(p_1,\ldots,p_n) - p\cdot H\Big(\frac{p_{n-1}}{p}, \frac{p_n}{p}\Big). \end{aligned}$$

Let δ be the depth of the leaves $n-1$ and n. Since $H(p_{n-1}/p, p_n/p) \le 1$, by Exercise 10.4, the average depth of t is

$$\begin{aligned} d^* - (\delta-1)p + \delta(p_{n-1}+p_n) &= d^* + p \\ &\ge H(p_1,\ldots,p_n) + p\left(1 - H\Big(\frac{p_{n-1}}{p}, \frac{p_n}{p}\Big)\right) \\ &\ge H(p_1,\ldots,p_n). \end{aligned}$$

(ii) $\displaystyle\sum_{1\le j\le l} n_j 2^{-j} = \sum_{1\le i\le n} 2^{-l_i} \le \sum_{1\le i\le n} p_i = 1.$

(iii) We proceed by induction on j. Initially, when $j = 1$, we have $2^j = 2$ nodes of depth 1, and (ii) implies that $n_1 = 2\cdot n_1 2^{-1} \le 2\sum_{1\le k\le l} n_k 2^{-k} \le 2$. Now we assume that $j \ge 1$ and the invariant holds before the jth pass through step 3. After removing the subtrees of n_j nodes of depth j, there remain $2^j - n_1 2^{j-1} - \cdots - n_{j-1}\cdot 2 - n_j$ internal nodes of depth j. Each of them has two children, and thus there are

$$\begin{aligned} 2^{j+1} - n_1 2^j - \cdots - n_j\cdot 2 &= 2^{j+1}\left(1 - \sum_{1\le k\le l} n_k 2^{-k}\right) + n_{j+1} + n_{j+2}2^{-1} + \cdots \\ &\quad + n_l 2^{j+1-l} \\ &\ge n_{j+1} \end{aligned}$$

nodes of depth $j+1$, by (ii), and the invariant holds before the $(j+1)$st pass through step 3 as well. Thus after the loop 2, t is a binary tree with at least n leaves, and after step 4 there remain precisely n leaves with weights $p_1,\ldots,p_n$. Step 5 does not remove any leaves, and hence step 6 indeed returns a stochastic mobile with leaf weights $p_1,\ldots,p_n$.

(iv) By construction, the depth of leaf i after step 4 is l_i, and hence the average depth is

$$\sum_{1\le i\le n} l_i p_i < \sum_{1\le i\le n} (-\log p_i + 1)p_i = H(p_1,\ldots,p_n) + 1.$$

Since the average depth does not increase when removing edges in step 5, the claim follows.

10.7 (ii) At an internal node v of weight $p(v)$, we multiply two polynomials whose product has degree $p(v)n$, at a cost of at most $\mathsf{M}(p(v)n) = p(v)nS(p(v)n) \le p(v)nS(n) = p(v)\mathsf{M}(n)$. By (i), the overall cost is at most $\sum_v p(v)\,\mathsf{M}(n) = d\,\mathsf{M}(n)$, where the sum is over all internal nodes v of t. By Exercise 10.5, we may choose t such that $d < H(p_0,\ldots,p_{r-1}) + 1$, and the claim follows.

10.8 Induction on i shows that $\lambda(M_{i,j}) \le 2^i l$. The cost for computing $M_{i,j}$ from its two children is $O(\mathsf{M}(2^i l))$ word operations. There are 2^{k-i} nodes at level i and $k = \log r$ levels, and the claim follows as in Exercise 10.3.

10.11 For multipoint evaluation, we may precompute the $M_{i,j}$ and the inverses modulo x^{2^i} of their reversals, as in Exercise 10.9. Then the cost for one remainder computation modulo $M_{i,j}$ drops to $2\mathsf{M}(2^i) + O(2^i)$ ring operations, and summing over all i,j gives at most $(2\mathsf{M}(n) + O(n))\log n$. In the interpolation algorithm 10.11, steps 1 and 2 are precomputation steps, and by Theorem 10.10, the cost for step 3 is $(\mathsf{M}(n) + O(n))\log n$.

10.12 (i) ALGORITHM 10.28 CRA over $F[x]$ for two moduli.

Input: Coprime monic $m_1, m_2 \in F[x]$ and $v_1, v_2 \in F[x]$ such that $\deg v_1 < \deg m_1 \le n$ and $\deg v_2 < \deg m_2 \le n$.

Output: The unique polynomial $f \in F[x]$ of degree less than $\deg m_1 + \deg m_2$ satisfying $f \equiv v_1 \bmod m_1$ and $f \equiv v_2 \bmod m_2$.

1. **call** the fast EEA to compute $s,t \in F[x]$ such that $sm_1 + tm_2 = 1$, $\deg s < \deg m_2$, and $\deg t < \deg m_1$.
2. $b_1 \longleftarrow v_1 t \text{ rem } m_1, \quad b_2 \longleftarrow v_2 s \text{ rem } m_2$
3. **return** $f = b_1 m_2 + b_2 m_1$

Correctness of the algorithm was shown in Section 5.4. The cost for step 1 is $O(\mathsf{M}(n)\log n)$ field operations, by Theorem 11.7, and steps 2 and 3 take only $O(\mathsf{M}(n))$.

(ii) The idea for a recursive approach is to compute the two interpolating polynomials for the first and the second half of the points, respectively, and construct

f from them using (i). If $T(n)$ denotes the cost of the algorithm, then the cost for the two recursive calls is $2T(n/2)$, plus $O(\mathsf{M}(n)\log n)$ for constructing f, by (i). Together with $T(1) = 0$, Lemma 8.2 yields $T(n) \in O(\mathsf{M}(n)\log^2 n)$. This is slower by a factor of $\log n$ than Algorithm 10.11.

10.13 "(i) $\Longrightarrow$ (iii)": Suppose that χ is an isomorphism, and let $0 \le i, j < r$ with $i \ne j$. Since χ is surjective, there exists a polynomial $f \in R[x]$ such that $f \equiv 1 \bmod m_i$ and $f \equiv 0 \bmod m_k$ for $k \ne i$. In particular, the latter is true for $k = j$, and we may write $f = s_{ij}m_j$ for some $s_{ij} \in R[x]$. Thus $s_{ij}m_j \equiv 1 \bmod m_i$, or equivalently, $s_{ij}m_j + t_{ij}m_i = 1$ for some $t_{ij} \in R[x]$.

"(iii) $\Longrightarrow$ (ii)" follows from $\prod_{i,j}(s_{ij}m_j + t_{ij}m_i) = 1$.

"(ii) $\Longrightarrow$ (i)": The assumption implies that $s_i(m/m_i) \equiv 1 \bmod m_i$ for all i. To show that χ is surjective, let $v_0, \ldots, v_{r-1} \in R[x]$ be arbitrary. Then the polynomial

$$f = \sum_{1 \le i \le r} v_i s_i \frac{m}{m_i}$$

satisfies $f \equiv v_i \bmod m_i$ for $0 \le i < r$, and hence $\chi(f \bmod m) = (v_0 \bmod m_0, \ldots, v_{r-1} \bmod m_{r-1})$. For the injectivity, we assume that $f \in R[x]$ satisfies $f \equiv 0 \bmod m_i$ for all i, say $f = u_i m_i$. Multiplying (ii) by f yields

$$f = \sum_{0 \le i < r} f s_i \frac{m}{m_i} = m \sum_{0 \le i < r} u_i s_i.$$

Finally, the equivalence "(iii) $\Longleftrightarrow$ (iv)" is Exercise 6.15.

10.15 We proceed by induction on k. If $k = 0$, then $\deg f < n = \deg m_0$ implies that $f = f \operatorname{rem} m_0$. If $k \ge 1$, then we may assume inductively that the results of the recursive calls in steps 3 and 4 are correct. Let $0 \le i < r/2$ and $q_0 = f \operatorname{quo} M_{k-1,0}$. Then $m_i \mid M_{k-1,0}$ and $f = q_0 M_{k-1,0} + r_0 \equiv f_0 \bmod m_i$, and we find $f \operatorname{rem} m_i = f_0 \operatorname{rem} m_i$. The proof for $r/2 \le i < r$ is similar.

(In the 1999 edition, f_0 and f_1 were called r_0 and r_1, respectively.)

10.16 Correctness follows as in the proof of Theorem 10.10. Let $T(k,n)$ denote the cost of the algorithm. Then the cost for step 1 is 0, the two recursive calls in steps 2 and 3 take $T(k-1, \deg M_{k-1,0})$ and $T(k-1, \deg M_{k-1,1})$, respectively, and the cost for step 4 is at most $2\mathsf{M}(n) + O(n)$. The cost estimate follows from $T(0,n) = 0$ by induction on $k = \log r$.

10.17 Let $d = \deg m_i$ for all i, so that $n = rd = 2^k d$.

(i) As in Exercise 10.3, the degree of $M_{i,j}$ is $2^i d$, and the cost for one remainder computation modulo $M_{i,j}$ is $\mathsf{D}(2^i d) \in 5\mathsf{M}(2^i d) + O(2^i d)$. Thus the cost per node at level i is $10\mathsf{M}(2^{i-1}d) + O(2^i d)$. There are 2^{k-i} nodes at level i and $k = \log r$ levels, and summing up shows that the overall cost for Algorithm 10.20 is at most $(10\mathsf{M}(n/2) + O(n))\log r$. The first claim now follows from Exercise 10.3.

If $T(n)$ denotes the cost of Algorithm 10.20, then we have $T(d) = 0$ and $T(n) = 2T(n/2) + 2\mathsf{M}(n/2+1)$, and Exercise 8.34 and Lemma 8.2 show that $T(n)$ is at most $(\mathsf{M}(n) + O(n)) \log r$.

(ii) We have $\sum_{0 \le i < r} \deg m_i^2 = 2n$, and hence (i) implies that the cost for step 1 is at most $(11\mathsf{M}(n) + O(n)) \log r$ operations in R. Step 2 takes $r\mathsf{D}(d)$ or $5\mathsf{M}(n) + O(n)$ operations, and the cost for step 3 is $r(24\mathsf{M}(d) + O(d)) \log d$ field operations, by Theorem 11.7.

(iii) The cost for step 1 is $(\frac{1}{2}\mathsf{M}(n) + O(n)) \log r$, by Exercise 10.3. Part (ii) of this Exercise gives the cost for step 2, and step 3 takes another $(\mathsf{M}(n) + O(n)) \log r$, by (i). The claim follows by summing up and using $(\frac{1}{2} + 11 + 1) \log r + 5 \le 24 \log r$ if $r \ge 2$.

10.21 ALGORITHM 10.29 Small primes modular quotient in $\mathbb{Z}[x]$.
Input: Nonzero $a, b \in \mathbb{Z}[x]$ with $\deg b < \deg a = n$ and $\|a\|_\infty \le A$.
Output: The quotient $a/b \in \mathbb{Z}[x]$ if $b \mid a$, and otherwise "FAIL".

1. $B \longleftarrow (n+1)^{1/2} 2^n A$
   if $\|b\|_\infty > B$ or $\mathrm{lc}(b) \nmid \mathrm{lc}(a)$ **then return** "FAIL"
2. $r \longleftarrow \lceil \log(2\,\mathrm{lc}(b) B + 1) \rceil$
   choose primes $2 < p_1 < p_2 < \cdots < p_r < 2r \ln r$
   $S \longleftarrow \{1 \le i \le r : p_i \nmid \mathrm{lc}(b)\}$
3. **for** all $i \in S$ compute a rem p_i and b rem p_i
4. **for** all $i \in S$ **do**
   if b does not divide a modulo p_i **then return** "FAIL"
   else compute $q_i \in \mathbb{Z}[x]$ with $a \equiv q_i b \bmod p_i$, $\deg q_i \le \deg a - \deg b$, and $\|q_i\|_\infty < p_i/2$
5. compute $q \in \mathbb{Z}[x]$ with $\deg q \le \deg a - \deg b$, $\|q_i\|_\infty < \frac{1}{2} \prod_{1 \le i \le r} p_i$, and $q \equiv q_i \bmod p_i$ for all $i \in S$
6. **if** $\|q\|_1 \|b\|_1 \le B$ **then return** q **else return** "FAIL"

For the correctness proof, let $m = \prod_{i \in S} p_i$. Then $m > 2B$ and $a \equiv qb \bmod m$. If $b \mid a$, then clearly $q = b/a$, and the algorithm returns q, by Corollary 6.33. Conversely, if $\|q\|_1 \|b\|_1 \le B$, then $\|qb\|_\infty \le \|qb\|_1 \le \|q\|_1 \|b\|_1 \le B < m/2$, and the congruence $a \equiv qb \bmod m$ is in fact an equality.

We have $\log p_i \in O(\log r)$ for all i and $\log m \in O(r \log r)$. Using the integer variants of Algorithms 10.16 and 10.22 for each coefficient of a, b, and q, respectively, steps 2, 3, and 5 take $O(n\mathsf{M}(r \log r) \log r)$ word operations, by Theorems 10.24 and 10.25. Step 4 takes $O(\mathsf{M}(n))$ additions and multiplications plus one inversion in $\mathbb{F}_{p_i}$ for each i, in total $O(r\mathsf{M}(\log r)(\mathsf{M}(n) + \mathrm{loglog} r))$ word operations, by Corollary 11.10. The time estimate now follows from $r \in O(\log B)$.

Chapter 11

11.3 This is wrong, as the example $f = x^2$, $g = x$, $f^* = x^3 + x$, $g^* = x^2 + 1$ for $k = 1$ shows.

11.5 Let $r_i^* \in \mathbb{Z}[x]$ for $0 \le i \le \ell$ denote the primitive associates of the remainders in the Euclidean Algorithm. Then

$$\operatorname{lc}(r_i^*)^{n_{i-1}-n_i+1} r_{i-1}^* = q_i^* r_i^* + \rho_{i+1}^* r_{i+1}^*,$$

where $\rho_{i+1}^* = \operatorname{cont}(\operatorname{lc}(r_i^*)^{n_{i-1}-n_i+1} r_{i-1}^* \operatorname{rem} r_i^*) \in \mathbb{Z}$, so that

$$Q_i^* = \begin{pmatrix} 0 & 1 \\ (\rho_{i+1}^*)^{-1} \operatorname{lc}(r_i^*)^{n_{i-1}-n_i+1} & -(\rho_{i+1}^*)^{-1} q_i^* \end{pmatrix}.$$

We replace Q_i by Q_i^* in step 9, and step 6 by

6. $a_{i-1} \longleftarrow \operatorname{lc}(r_i^*)^{1+n_{i-1}-n_i} r_{i-1}^*$, $\quad q_i^* \longleftarrow a_{i-1} \operatorname{quo} r_i^*$,
   $\rho_{i+1}^* \longleftarrow \operatorname{cont}(a_{i-1} \operatorname{rem} r_i^*)$, $\quad r_{i+1}^* \longleftarrow \operatorname{pp}(a_{i-1} \operatorname{rem} r_i^*)$, $\quad n_{i+1} \longleftarrow \deg r_{i+1}^*$

The matrix Q_i^* has rational entries in general.

11.6 The standard approach, as in the proof of Theorem 11.5, uses 10 polynomial multiplications where the product polynomial has degree at most $k < \kappa$, each of them taking essentially three κ-point FFTs, in total 30 FFTs plus $O(\kappa)$ operations. In the alternative approach, we have nine κ-point FFTs for evaluating q_j and the entries of R and S, plus another four for the interpolation of the entries of the product matrix, in total only 13 FFTs plus $O(\kappa)$ operations.

11.7 ALGORITHM 11.19 Fast Extended Euclidean Algorithm.
Input: $r_0, r_1 \in F[x]$ monic, $n_0 = \deg r_0 > n_1 = \deg r_1$, $k \in \mathbb{N}$ with $n_0/2 \le k \le n_0$.
Output: $h = \eta(k) \in \mathbb{N}$, $R_h \in F[x]^{2\times 2}$ as in (1), and $\begin{pmatrix} r_h \\ r_{h+1} \end{pmatrix} = R_h \begin{pmatrix} r_0 \\ r_1 \end{pmatrix}$.

1. **if** $r_1 = 0$ or $k < n_0 - n_1$ **then return** 0, $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, and $\begin{pmatrix} r_0 \\ r_1 \end{pmatrix}$
2. $d \longleftarrow \lfloor k/2 \rfloor$
3. $a_0 = r_0 \upharpoonright 2d$, $\quad a_1 \longleftarrow r_1 \upharpoonright (2d - (n_0 - n_1))$
   call the algorithm recursively with input a_0, a_1 and d, giving $j - 1 = \eta(d)$, $R = Q_{j-1} \cdots Q_1$, and $\begin{pmatrix} a_{j-1} \\ a_j \end{pmatrix} = R \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$.
4. $\begin{pmatrix} r_{j-1} \\ r_j \end{pmatrix} \longleftarrow \begin{pmatrix} a_{j-1} x^{n_0-2d} \\ a_j x^{n_0-2d} \end{pmatrix} + R \begin{pmatrix} r_0 - a_0 x^{n_0-2d} \\ r_1 - a_1 x^{n_0-2d} \end{pmatrix}$
   $\begin{pmatrix} n_{j-1} \\ n_j \end{pmatrix} \longleftarrow \begin{pmatrix} \deg r_{j-1} \\ \deg r_j \end{pmatrix}$

5. **if** $r_j = 0$ or $k < n_0 - n_j$ **then return** $j-1$, R, and $\begin{pmatrix} r_{j-1} \\ r_j \end{pmatrix}$

6. $q_j \longleftarrow r_{j-1}$ quo r_j, $\quad \rho_{j+1} \longleftarrow \mathrm{lc}(r_{j-1}$ rem $r_j)$,
$r_{j+1} \longleftarrow (r_{j-1}$ rem $r_j)\rho_{i+1}^{-1}$, $\quad n_{j+1} \longleftarrow \deg r_{j+1}$

7. $d^* \longleftarrow k - (n_0 - n_j)$

8. $a_j^* \longleftarrow r_j \restriction 2d^*$, $\quad a_{j+1}^* \longleftarrow r_{j+1} \restriction (2d^* - (n_j - n_{j+1}))$
call the algorithm recursively with input a_j^*, a_{j+1}^* and d^*, giving $h - j = \eta(d^*)$, $S = Q_h \cdots Q_{j+1}$, and $\begin{pmatrix} a_h^* \\ a_{h+1}^* \end{pmatrix} = S \begin{pmatrix} a_j^* \\ a_{j+1}^* \end{pmatrix}$

9. $Q_j \longleftarrow \begin{pmatrix} 0 & 1 \\ \rho_{j+1}^{-1} & -q_j\rho_{j+1}^{-1} \end{pmatrix}$
return h, SQ_jR, and $\begin{pmatrix} a_h^* x^{n_j - 2d^*} \\ a_{h+1}^* x^{n_j - 2d^*} \end{pmatrix} + S \begin{pmatrix} r_j - a_j^* x^{n_j - 2d^*} \\ r_{j+1} - a_{j+1}^* x^{n_j - 2d^*} \end{pmatrix}$

The cost for step 4 is now essentially 4 multiplications of polynomials of degree about $k/2$ by polynomials of degree about k or $4\mathsf{M}(k) + O(k)$. Step 6 and the computation of Q_jR in step 9 take $O(k)$, and the computation of $S \cdot Q_jR$ in step 9 takes $4\mathsf{M}(k) + O(k)$, as in the proof of Theorem 11.5. Additionally, we have essentially 4 multiplications of polynomials of degree about $k/2$ or $2\mathsf{M}(k) + O(k)$ for the computation of r_h, r_{h+1} in step 9, and the claim follows.

A variant of this algorithm is given by Brent, Gustavson & Yun (1980).

11.9 (i) We first call Algorithm 11.4 with input f, g, and $k = \deg f - e_1$. Then we compute $(r_h, r_{h+1})^T = R_h \cdot (f,g)^T$. The polynomial r_h is the remainder of degree e_1. Since $e_2 < e_1$, the remainder of degree e_2 in the EEA of f and g is equal to the remainder of degree e_2 in the EEA of r_h and r_{h+1}, and we can proceed recursively. The overall cost is

$$O\Big(\mathsf{M}(n)\log n + \mathsf{M}(e_1)\log e_1 + \cdots + \mathsf{M}(e_{d-1})\log e_{d-1}\Big)$$

arithmetic operations in F, and the claim follows from $e_1 + \cdots + e_{d-1} \le n$ and the superlinearity of M.

(ii) We modify Algorithm 6.59 so as to compute only the required remainders. We can compute $f(x,u)$ and $g(x,u)$ for all $u \in U$ in step 2 using $O(n\mathsf{M}(nd)\log(nd))$ arithmetic operations, by Corollary 10.8. By (i), the cost for computing only the remainders of the required degrees in step 2 is $O(nd\,\mathsf{M}(n)\log n)$ operations. In step 3, we again interpolate only the remainders of the required degrees. The cost per coefficient is $O(\mathsf{M}(nd)\log(nd))$ arithmetic operations, by Corollaries 10.12 and 11.6, and there are $e_1 + \cdots + e_d \le n$ coefficients. The claim now follows by adding up costs.

Chapter 12

12.1

$$
\begin{aligned}
U_1 &= P_1 + P_2 = A_{11}B_{11} + A_{12}B_{21},\\
U_2 &= P_1 + P_6 = A_{11}B_{11} + S_2T_2 = A_{11}B_{11} + (S_1 - A_{11})(B_{22} - T_1)\\
&= A_{11}B_{11} + (-A_{11} + A_{21} + A_{22})(B_{11} - B_{12} + B_{22})\\
&= A_{11}B_{12} - A_{11}B_{22} + A_{21}B_{11} - A_{21}B_{12} + A_{21}B_{22}\\
&\quad + A_{22}B_{11} - A_{22}B_{12} + A_{22}B_{22},\\
U_3 &= U_2 + P_7 = U_2 + S_3T_3 = U_2 + (A_{11} - A_{21})(B_{22} - B_{12})\\
&= A_{21}B_{11} + A_{22}B_{11} - A_{22}B_{12} + A_{22}B_{22},\\
U_4 &= U_2 + P_5 = U_2 + S_1T_1 = U_2 + (A_{21} + A_{22})(B_{12} - B_{11})\\
&= A_{11}B_{12} - A_{11}B_{22} + A_{21}B_{22} + A_{22}B_{22},\\
U_5 &= U_4 + P_3 = U_4 + S_4B_{22} = U_4 + (A_{12} - S_2)B_{22}\\
&= U_4 + (A_{11} + A_{12} - A_{21} - A_{22})B_{22}\\
&= A_{11}B_{12} + A_{12}B_{22},\\
U_6 &= U_3 - P_4 = U_3 - A_{22}T_4 = U_3 - A_{22}(T_2 - B_{21})\\
&= U_3 - A_{22}(B_{11} - B_{12} - B_{21} + B_{22})\\
&= A_{21}B_{11} + A_{22}B_{21},\\
U_7 &= U_3 + P_5 = U_3 + S_1T_1 = U_3 + (A_{21} + A_{22})(B_{12} - B_{11}) = A_{21}B_{12} + A_{22}B_{22}.
\end{aligned}
$$

12.4 (i) Let y be a new indeterminate. By Exercise 9.25, g has the Taylor expansion $g(x) = \sum_{0 \le i < n} g^{(i)}(y)(x-y)^i/i!$ around y in $R[y][x]$, and substituting $y = h_0$ and $x = h$ yields the claim.

(ii) h_0' is invertible modulo x^{n+k}, and Algorithm 9.3 computes its inverse at a cost of $O(\mathsf{M}(n))$ ring operations. Let $a = g^{(i)}(h_0)$. Then $g^{(i+1)}(h_0) = a' \cdot (h_0')^{-1} \equiv (a \bmod x^{n+k-i})'(h_0')^{-1} \bmod x^{n+k-i-1}$, and this computation takes another $O(\mathsf{M}(n))$ ring operations. We note that the latter congruence does not hold modulo x^{n+k-i} in general (see Exercise 9.24). Thus the precision is decreased by one in each step, and this is the reason why we start with the higher precision $n+k$ instead of n in step 2. Things get even more complicated in (v) below.

(iii) The cost is $O(k\mathsf{M}(n))$ ring operations for step 1, $O(m\mathsf{M}(n)\log n)$ for step 2, by Exercise 12.3, $O(k\mathsf{M}(n))$ for step 3, by (ii), and another $O(k\mathsf{M}(n))$ for step 4.

(iv) Letting $n/m \approx k \approx m\log n$ yields $m \approx (n/\log n)^{1/2}$ and a time bound of $O((n\log n)^{1/2}\mathsf{M}(n))$.

(v) We may assume that $h_0' \neq 0$, since otherwise h_0 is constant and all $g^{(i)}(h_0)$ can be computed in time $O(kn)$. So let $h_0' = x^d b$ with $0 \le d < m$ and $b(0) \neq 0$. The chain rule shows that $g^{(i)}(h_0)' = g^{(i+1)}(h_0)h_0'$ is divisible by x^d for all i. We

modify step 3 and perform all computations to slightly larger precision, as follows. Letting $a_i = g^{(i)}(h_0)$, we find that $a_i' = a_{i+1}h_0' = a_{i+1}x^d b$, and therefore

$$a_{i+1} \equiv b^{-1}(a_i \text{ rem } x^{n+(k-i)(d+1)})'/x^d \bmod x^{n+(k-i-1)(d+1)}.$$

Now $n+(k-i)(d+1) \le n+km \in O(n)$, and hence we can compute b^{-1} mod $x^{n+k(d+1)}$ with $O(\mathsf{M}(n))$ ring operations, and also a_{i+1} rem $x^{n+(k-i-1)(d+1)}$ from a_i rem $x^{n+(k-i)(d+1)}$. In step 2, we compute $g(h_0)$ rem $x^{n+k(d+1)}$, taking $O(k\mathsf{M}(n))$ ring operations, and the same bound is valid for step 3.

12.6 (i) $m_a = x^n$; (ii) $m_{x^n \bullet a} \mid m_a \mid x^n m_{x^n \bullet a}$.

12.7 (b) $\iff$ (c) is clear. So let $f = \sum_{0\le j\le d} f_j x^j$ be a characteristic polynomial of a. Then $r = \sum_{0\le j\le n} f_{d-j}x^j$ and $rh = \sum_{i\in\mathbb{N}} b_i x^i$ with $b_i = \sum_{0\le j\le i} f_{d-(i-j)}a_j$ for all i, where we set $f_j = 0$ if $j < 0$. Then $b_{d+i} = \sum_{i\le j\le d+i} f_{j-i}a_j = \sum_{0\le j\le d} f_j a_{i+j} = 0$ for all i if and only if f is a characteristic polynomial of a, and this shows the equivalence of (a) and (b).

12.8 (i) $a_0 = 0$, $a_1 = -1$, $a_2 = 0$, and $a_{i+3} = -a_{i+2} + a_i$ for $i \in \mathbb{N}$.

(ii) $a_0 = 1$, $a_1 = -2$, $a_2 = 2$, and $a_{i+3} = -a_{i+2} + a_i$ for $i \in \mathbb{N}$.

(iii) $a_0 = 0$, $a_1 = -1$, $a_2 = 1$, $a_3 = a_4 = -1$, and $a_{i+3} = -a_{i+2} + a_i$ for all $i \ge 2$.

12.9 The minimal polynomial is $x^2 - x - 1$, like for the Fibonacci sequence, and the first 20 elements are $1,3,4,7,11,18,29,47,76,123,199,322,521,843,1364,2207,3571,5778,9349,15127$.

12.10 (ii) Let m_a be the minimal polynomial of the sequence $a = (\tau(\beta_i))_{i\in\mathbb{N}}$. Then $m_a \mid m$, and since m is irreducible, we have either $m_a = 1$ or $m_a = m$. But $\tau(\beta^0) = 1$, so that $a \ne \mathbf{0}$, and this shows that $m_a = m$.

(iii) We first compute $\beta, \beta^2, \ldots, \beta^{2n-1}$ in polynomial representation. This takes $O(n\cdot\mathsf{M}(n))$ operations in F, by Corollary 11.8. Computing $\tau(\beta^i)$ for $0 \le i < 2n$ is for free, and computing the minimal polynomial of a takes $O(\mathsf{M}(n)\log n)$ field operations, by Theorem 12.10.

(iv) $m = x^3 - 3x^2 - 3x - 1$.

12.12 f is a characteristic polynomial of a^{**} if and only if $u^T f(A)A^i b = 0$ for $0 \le i < n$.

12.15 (ii) Let $d = \deg f \le n$. We define the F–linear map $\psi^*: F^n \times F^n \longrightarrow F^{\mathbb{N}}$ by $\psi^*(u,b) = (u^T A^i b)_{i\in\mathbb{N}}$, and let $M_f \subseteq F^{\mathbb{N}}$ be the submodule of all sequences annihilated by f. As in the proof of Lemma 12.16, we find that $\psi^*(F^n \times F^n) = M_f$, since $I, A, \ldots, A^{d-1}$ are linearly independent in $F^{n\times n}$. Then we let $\psi = \varphi^{-1} \circ \psi^*$, where $\varphi: F[x]/\langle f\rangle \longrightarrow M_f$ is the isomorphism of cyclic $F[x]$-modules from (9). Then ψ is F–bilinear and surjective, $g \bullet \psi^*(u,b) = \varphi((g \bmod f)\cdot\psi(u,b))$ for all $g \in F[x]$ and $u, b \in F^n$, and the claim follows, as in the proof of Lemma 12.16.

(iii) Let $e_j \in F^n$ be the jth unit vector for $1 \le j \le n$, $u = (u_1, \ldots, u_n)$ and $b = (b_1, \ldots, b_n)$ in F^n, and $h_{ij} \bmod f = \psi(e_i, e_j)$ for $1 \le i, j \le n$. Then the bilinearity of ψ yields

$$\psi(u,b) = \sum_{1 \le i,j \le n} u_i b_j h_{ij} \bmod f.$$

If we let

$$r = \operatorname{res}_x\Big(\sum_{1 \le i,j \le n} y_i z_j h_{ij},\ f \Big) \in F[y_1, \ldots, y_n, z_1, \ldots, z_n],$$

where the y_i and z_j are new indeterminates, then (ii) and Lemma 6.25 imply that $\psi(u,b)$ is a unit if and only if $r(u_1, \ldots, u_n, b_1, \ldots, b_n) = 0$. As in the proof of Lemma 12.17, we find that r is a nonzero polynomial of total degree at most $2n$, and the claim follows from Lemma 6.44.

12.16 (ii) Let $B \in F^{n \times r}$ be the matrix whose columns are $b_0, \ldots, b_{r-1}$. Then the b_i are linearly independent if and only if the rank of B equals r. This rank can be computed by Gaussian elimination (see Section 25.5), and this computation is the same whether we perform it over F or over K.

12.18 (i) Let $h = \gcd(f,g)$. Then we have $m \bullet (g \bullet a) = mg \bullet a = \mathbf{0}$ if and only if $f \mid mg$, which in turn is equivalent to $f/h \mid m$.

(ii) Let b^* be the initial value of b, f be the minimal polynomial of $(A^i b^*)_{i \in \mathbb{N}}$, and h the minimal polynomial of $(A^i b)_{i \in \mathbb{N}}$ in step 2. Then the invariants $f = gh$ and $b = g(A)b^*$ follow from (i) by induction. Now $b = g(A)b^* = 0$ in step 2 if and only if $f \mid g$, and the correctness follows.

(iii) We have $g \bullet a^{(k)} = (u_k A^i b)_{i \in \mathbb{N}}$, and by (i), the minimal polynomial m of this sequence is $g_k / \gcd(g, g_k)$. The second claim follows by induction, using Exercise 3.6.

(iv) For a fixed j, the probability that f_j divides all h_i is $q^{-k \deg f_j}$. By the Chinese Remainder Theorem, these r events are independent, and the stated formula for p_k follows.

(v) Let $s \in F[x]$. Similarly to the proof of Lemma 12.16, we have

$$s \bullet \psi^*(u) = \mathbf{0} \iff (s \bmod f) \cdot \psi(u) = 0 \iff f \mid sh \iff \frac{f}{\gcd(f,h)} \mid s,$$

proving the first claim. Let $h_k \bmod f = \psi(u_k)$ for all k. Then $g_k = f / \gcd(f, h_k)$. Inductively, we find that $\operatorname{lcm}(g_1, \ldots, g_k) = f$ if and only if $\gcd(h_1, \ldots, h_k, f) = 1$. Since the u_i are independent uniform random elements of F^n, the h_i are independent uniform random polynomials of degree less than $\deg f$. By (ii) and (iii), the algorithm terminates if and only if $f = g = \operatorname{lcm}(g_1, \ldots, g_k)$, and the claim follows from (iv).

(vi) The first claim follows by induction on r. Then

$$1-p_k = 1-\prod_{1\le j\le r}(1-q^{-k\deg f_j}) \le \sum_{1\le j\le r} q^{-k\deg f_j} = \sum_{i\ge 1} n_i q^{-ki}$$
$$\le \sum_{i\ge 1}\frac{q^i}{i}q^{-ki} = q^{1-k}\sum_{i\ge 0}\frac{(q^{1-k})^i}{i+1} \le q^{1-k}\sum_{i\ge 0}2^{-i} = 2q^{1-k}$$

if $k\ge 2$, where we used that $q\ge 2$, and hence

$$\sum_{k\ge 0}(1-p_k) \le 2+2\sum_{k\ge 2}q^{1-k} = 2+\frac{2}{q-1} \le 4.$$

Chapter 13

13.2 If all integrals exist, then for $k\in\mathbb{Z}$ we have

$$\widehat{(f*g)}(k) = \int_0^{2\pi}(f*g)(t)e^{-ikt}dt = \int_0^{2\pi}\int_0^{2\pi} f(s)g(t-s)e^{-ikt}ds\,dt$$
$$= \int_0^{2\pi} f(s)e^{-iks}\int_0^{2\pi} g(t-s)e^{-ik(t-s)}d(t-s)\,ds$$
$$= \left(\int_0^{2\pi} f(s)e^{-iks}ds\right)\cdot\widehat{g}(k) = \widehat{f}(k)\cdot\widehat{g}(k).$$

13.3 Using $\sin(t)=\frac{i}{2}(e^{-it}-e^{it})$, we find

$$f(t) = \frac{1}{2\pi}\left(\pi i e^{-it} - \pi i e^{it} + \frac{\pi i}{10}e^{-10it} - \frac{\pi i}{10}e^{10it}\right).$$

The uniqueness of the Fourier series (1) implies that $\widehat{f}(1) = -\widehat{f}(-1) = -\pi i$, $\widehat{f}(10) = -\widehat{f}(-10) = -\pi i/10$, and $\widehat{f}(k)=0$ for all other integers k.

13.4 (ii) We have $\beta_0 = 0$, and for $k\ne 0$, the kth Fourier coefficient is

$$\beta_k = \frac{1}{2\pi}\int_0^{2\pi} f(t)e^{-ikt}dt = \frac{1}{2\pi}\left(\int_0^{\pi} e^{-ikt}dt - \int_\pi^{2\pi} e^{-ikt}dt\right)$$
$$= \frac{i}{2\pi k}\left(e^{-ik\pi} - e^0 - e^{-ik\cdot 2\pi} + e^{-ik\pi}\right) = \frac{i}{\pi k}(e^{-ik\pi}-1)$$
$$= \begin{cases} \frac{-2i}{k\pi} & \text{if } k \text{ is odd;} \\ 0 & \text{if } k \text{ is even.}\end{cases}$$

So we have

$$f(t) = \sum_{\substack{k\in\mathbb{Z}\\ k \text{ odd}}}\frac{-2i}{k\pi}e^{ikt}.$$

Incidentally, this gives a method to compute π. We find

$$\begin{aligned} f(t) &= \beta_0 + \sum_{\substack{k>0 \\ k \text{ odd}}} (\beta_k e^{ikt} + \beta_{-k} e^{-ikt}) = \sum_{\substack{k>0 \\ k \text{ odd}}} \frac{-2i}{k\pi}(e^{ikt} - e^{-ikt}) + 0 \\ &= \sum_{\substack{k>0 \\ k \text{ odd}}} \frac{-2i}{k\pi}((\cos(kt) + i\sin(kt)) - (\cos(kt) - i\sin(kt))) \\ &= \sum_{\substack{k>0 \\ k \text{ odd}}} \frac{4}{k\pi}\sin(kt), \end{aligned}$$

or in other words,

$$f(t) = \frac{4}{\pi}(\sin(t) + \frac{1}{3}\sin(3t) + \frac{1}{5}\sin(5t) + \cdots).$$

From $f(\pi/2) = 1$ we may deduce that

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots = \frac{\pi}{4}.$$

This equation, already known to Leibniz in 1673, is not a practical way of calculating the digits of π (Section 4.6).

13.6 (i) We have

$$\begin{aligned} \widehat{g}(k) &= \sum_{0 \le j < 4n} g(j) e^{-2\pi i jk/4n} \\ &= \sum_{0 \le j < n} \left(g(2j+1) e^{-\pi ik(2j+1)/2n} + g(4n-2j-1) e^{-\pi ik(4n-2j-1)/2n} \right) \\ &= \sum_{0 \le j < n} f(j) \left(e^{-\pi ik(2j+1)/2n} + e^{\pi ik(2j+1)/2n} \right) \\ &= 2 \sum_{0 \le j < n} f(j) \cos \frac{\pi k(2j+1)}{2n} \end{aligned}$$

for $k \in \mathbb{Z}$. The claimed symmetry properties follow from those of the cosine.

(ii) Part (i) implies in particular that $\widehat{g}(2n) = -\widehat{g}(0)$ and $\widehat{g}(n) = \widehat{g}(3n) = 0$. The claim follows from the symmetry properties together with the inversion formula

$$g(j) = \frac{1}{4n} \sum_{0 \le k < 4n} \widehat{g}(k) e^{2\pi ijk/4n}$$

for the Discrete Fourier Transform.

(j and n were called n and N, respectively, in the 1999 edition.)

Chapter 14

14.1 (i) For each $a \neq \pm 1$, both a and its inverse $a^{-1} \neq a$ occur exactly once in the product and cancel each other. Thus $\prod_{a \in \mathbb{F}_q^\times} a = \prod_{a=\pm 1} a = -1$; this is also valid for even q, where $1 = -1$.

(ii) If n is not prime, then it has a prime divisor $p < n$ which divides $(n-1)!$ but does not divide -1.

14.2 f has no proper divisors of degree at most 2, and hence it is irreducible.

14.3 The factorization pattern of f is $(1,1,1,2,2,4,6)$.

14.6 (i) An irreducible factor of f divides $\prod_{a \leq d < b}(x^{q^d} - x)$ if and only if its degree divides some number in the interval $\{a, a+1, \ldots, b-1\}$, by Theorem 14.2, and the claim follows since f is squarefree.

(ii) We have $x^{q^b} - x^{q^{b-d}} = (x^{q^d} - x)^{q^{b-d}}$, so that

$$\prod_{a \leq d < b} (x^{q^b} - x^{q^{b-d}}) = \left(\prod_{a \leq d < b} (x^{q^d} - x) \right)^{q^e},$$

where $e = \sum_{a \leq d < b}(b-d) = \sum_{1 \leq d \leq b-a} d = (b-a+1)(b-a)/2$. Thus the gcd is the same as in (i), by (i) and the squarefreeness of f.

(iii) ALGORITHM 14.56 Interval distinct-degree factorization.

Input: A monic squarefree polynomial $f \in \mathbb{F}_q[x]$ of degree $n \geq 1$, where q is a prime power, and integers $1 = c_0 < c_1 < \cdots < c_k = n+1$.

Output: The monic polynomials $g_1, \ldots, g_s \in \mathbb{F}_q[x]$ such that $s \leq k$, each g_j is the product of all monic irreducible factors of f whose degree is in the interval $I_j = \{c_{j-1}, \ldots, c_j - 1\}$, and $g_s \neq 1$.

1. $h_0 \longleftarrow x, \quad f_0 \longleftarrow f, \quad i \longleftarrow 0, \quad j \longleftarrow 0$
   repeat
2. $\{$ $i = c_j - 1$ and f_j is the product of all irreducible factors of f of degree at least c_j $\}$
   $j \longleftarrow j+1, \quad u \longleftarrow 1$
   while $i < c_j - 1$ **do**
3. $\{$ $h_i = x^{q^i}$ rem f and $u = \prod_{c_{j-1} \leq d \leq i}(x^{q^d} - x)$ rem f $\}$
   $i \longleftarrow i+1$
   call the repeated squaring algorithm 4.8 in $R = \mathbb{F}_q[x]/\langle f \rangle$ to compute $h_i = h_{i-1}^q$ rem f
   $u \longleftarrow u \cdot (h_i - x)$ rem f
4. $g_j \longleftarrow \gcd(u, f_{j-1}), \quad f_j \longleftarrow \dfrac{f_{j-1}}{g_j}$
5. **until** $f_j = 1$

6. $s \longleftarrow j$
   return $(g_1, \cdots, g_s)$

This blocking idea becomes really useful with more efficient ways to compute the required values of u, as in von zur Gathen & Shoup (1992) and Kaltofen & Shoup (1998).

14.8 When q is even, then squaring is an automorphism of $\mathbb{F}_q$, and there are no nonsquares. Otherwise, if q is odd then $(ab)^{(q-1)/2} = a^{(q-1)/2} b^{(q-1)/2} = (-1) \cdot (-1) = 1$, by Lemma 14.7.

14.11 (ii) For $n \in \mathbb{Z}$, we have $a^{kn} = 1$ if and only if $\operatorname{ord} a$ divides kn, or equivalently, $\operatorname{ord} a / \gcd(k, \operatorname{ord} a)$ divides n. See also Exercise 8.13.

(iii) Let $a \in \mathbb{F}_q^\times$, and assume that $\gcd(k, q-1) = 1$. By Lagrange's theorem (or Fermat's little theorem), we have $\operatorname{ord} a \mid q - 1 = \mathbb{F}_q^\times$. Thus $\gcd(k, \operatorname{ord} a) = 1$. Then (ii) shows that $\operatorname{ord}(\sigma_k(a)) = \operatorname{ord} a$. In particular, the only element with $\operatorname{ord}(\sigma_k(a)) = 1$ is $a = 1$, so that $\ker \sigma_k = \{1\}$ and σ_k is injective. Now $\mathbb{F}_q^\times$ is a finite set, so that σ_k is surjective as well.

Conversely, let σ_k be an automorphism and a a generator of $\mathbb{F}_q^\times$, which exists by Exercise 8.16. Then $\sigma_k(a)$ is also a generator of $\mathbb{F}_q^\times$, so that $\operatorname{ord}(\sigma_k(a)) = \operatorname{ord} a = q - 1$, and (ii) implies that $\gcd(k, q-1) = 1$.

(iv) Let $k = gk^*$. Then $\gcd(k^*, q-1) = 1$, (iii) implies that σ_{k^*} is an automorphism, and the claims follow from $\sigma_k = \sigma_g \circ \sigma_{k^*}$.

14.13 If $p = 2$, then $a = 1$ is its only square root. If p is odd, apply the equal-degree factorization algorithm 14.10 to the squarefree polynomial $x^2 - a \in \mathbb{F}_p[x]$. You find $(\pm 1111)^2 \equiv 1005 \bmod 2591$.

14.15 The degree of the smaller factor is at most $\lfloor n/2 \rfloor$, and the claim follows from

$$\begin{aligned}
\sum_{i \geq 0} (d \log q + \log(n2^{-i})) \mathsf{M}(\lfloor n2^{-i} \rfloor) &\leq (d \log q + \log n) \sum_{i \geq 0} \mathsf{M}(\lfloor n2^{-i} \rfloor) \\
&\leq (d \log q + \log n) \mathsf{M}(n) \sum_{i \geq 0} 2^{-i} \\
&\leq 2(d \log q + \log n) \mathsf{M}(n),
\end{aligned}$$

where we used the superlinearity of M.

14.16 (i) The first claim follows from $T_m(T_m + 1) = T_m^2 + T_m = T_m(x^2) + T_m$. Now let $\alpha \in \mathbb{F}_q$. By Fermat's little theorem, α is a root of $x^{2^m} + x$, so that it is a root of T_m or of $T_m + 1$, and hence either $T_m(\alpha) = 0$ or $T_m(\alpha) = 1$. Both T_m and $T_m + 1$ have degree 2^{m-1}, and hence each of them has precisely 2^{m-1} roots.

(ii) The field $R_i = \chi_i(R)$ is isomorphic to $\mathbb{F}_{q^d} = \mathbb{F}_{2^{kd}}$, and the first claim follows from (i) with k replaced by kd. By the Chinese Remainder Theorem, $T_{kd}(\alpha)$ is in $\mathbb{F}_2$ if and only if the $\chi_i(T_{kd}(\alpha))$ are equal for all i, so that either all are 1 or all are 0. Since the $\chi_i(T_{kd}(\alpha))$ are independent random variables if α is chosen uniformly at random, the probability that this happens is $2 \cdot 2^{-r}$, by (i).

(iii) The modified algorithm fails if $T_{kd}(a \bmod f) \in \mathbb{F}_2$, and (ii) gives the error probability. We can compute $T_{kd}(a)$ rem f by computing all a^{2^i} rem f for $0 \le i < kd$, taking $kd-1$ squarings modulo f or $O(kd\,\mathsf{M}(n))$ operations in $\mathbb{F}_q$, and adding up the results, using at most $(kd-1)n$ additions in $\mathbb{F}_q$. Since $k = \log q$, this yields the same cost estimate as for step 3 of the original algorithm, and the claim follows.

(iv) Let $\mathcal{B}_i = \chi_i(\mathcal{B}) \subseteq R_i$. Then $\mathcal{B}_i \cong \mathbb{F}_q$, and hence $\chi_i(T_k(\alpha)) = T_k(\chi_i(\alpha)) \in \mathbb{F}_2$ for all $\alpha \in \mathcal{B}$, by (i). The probability estimate follows as in (ii).

(v) The estimate of the failure probability follows from (iv). Similarly as in (iii), the time for computing $T_k(a)$ rem f is $O(k\,\mathsf{M}(n))$ operations in $\mathbb{F}_q$, which is the same estimate as for step 6 of Algorithm 14.31.

14.17 In ths 1999 edition, the text of the exercise contains several typos, and we first give a corrected version of it.

Let q be an odd prime power and $f \in \mathbb{F}_q[x]$ squarefree of degree n with $r \ge 2$ irreducible factors $f_1, \ldots, f_r$ of degree $d = n/r$. We let $R, R_1, \ldots, R_r$ and the Chinese remainder isomorphism $\chi = \chi_1 \times \cdots \times \chi_r : R \longrightarrow R_1 \times \cdots \times R_r$ be as in Section 14.3. The **norm** on $R_i \cong \mathbb{F}_{q^d}$ is defined by $N(\alpha) = \alpha\alpha^q\alpha^{q^2}\cdots\alpha^{q^{d-1}} = \alpha^{(q^d-1)/(q-1)}$, and we use the same formula to define the norm on R.

(i) Let $\alpha \in R^\times$ be a uniform random element, $\beta = N(\alpha)$, and $1 \le i \le r$. Show that $\chi_i(\beta)$ is a root of $x^{q-1}-1$, and conclude that $\chi_i(\beta)$ is a uniform random element in $\mathbb{F}_q^\times$. Hint: N is a homomorphism of multiplicative groups.

(ii) Provided that $q > r$, what is the probability that the $\chi_i(\beta)$ are distinct for $1 \le i \le r$? Prove that this probability is at least $1/2$ if $q - 1 \ge r^2$.

(iii) For $u \in \mathbb{F}_q$, let $\pi(u) = u^{(q-1)/2}$, so that $\pi(u) \in \{-1, 0, 1\}$, $\pi(u) = 0$ if and only if $u = 0$, and $\pi(u) = -1$ if and only if u is a nonsquare. Moreover, let $u, v \in \mathbb{F}_q$ be distinct. Prove that for a uniformly random $t \in \mathbb{F}_q$, we have $\pi(u+t) \ne \pi(v+t)$ with probability at least $1/2$. Hint: The map $t \longmapsto (u+t)/(v+t)$ if $t \ne -v$ and $-v \longmapsto 1$ is a bijection of $\mathbb{F}_q$.

(iv) Consider the following variant of Algorithm 14.8, due to Rabin (1980b).

ALGORITHM 14.54 Equal-degree splitting.

Input: A squarefree monic reducible polynomial $f \in \mathbb{F}_q[x]$ of degree n, where q is an odd prime power, a divisor $d < n$ of n, so that all irreducible factors of f have degree d, and $a \in \mathbb{F}_q[x]$ of degree less than n with $\chi_i(a \bmod f) \in \mathbb{F}_q$ for all i.

Output: A proper monic factor $g \in \mathbb{F}_q[x]$ of f, or "failure".

1. $g_1 \longleftarrow \gcd(a, f)$
   if $g_1 \ne 1$ and $g_1 \ne f$ **then return** g_1
2. choose $t \in \mathbb{F}_q$ at random
3. **call** the repeated squaring algorithm 4.8 in $R = \mathbb{F}_q[x]/\langle f \rangle$ to compute $b = (a+t)^{(q-1)/2}$ rem f

4. $g_2 \longleftarrow \gcd(b-1, f)$
   if $g_2 \neq 1$ and $g_2 \neq f$ **then return** g_2 **else return** "failure"

Use (iii) to prove that the failure probability of the algorithm is at most $1/2$ if $a \notin \mathbb{F}_q$.

(v) Use the algorithm from (iv) as a subroutine to create a recursive algorithm for equal-degree factorization, which has the same input specification as the above algorithm and outputs all irreducible factors of f. Prove that the algorithm never halts if $\chi_i(a \bmod f) = \chi_j(a \bmod f)$ for some $i \neq j$, and that otherwise, if all $\chi_i(a \bmod f)$ are distinct elements of $\mathbb{F}_q$, the probability for its recursion depth to be more than $k = 1 + \lceil 2\log_2 r \rceil$ is at most $1/2$. Conclude that in the latter case, the number of operations in $\mathbb{F}_q$ is $O(\mathsf{M}(n)\log(qn)\log r)$.

(vi) Now we first compute $a = c^{(q^d-1)/(q-1)} \bmod f$ for a uniform random polynomial $c \in \mathbb{F}_q[x]$ of degree less than n, and then call the algorithm from (v) for that value of a and stop the recursion at depth k. We assume that $q - 1 \geq r^2$. Prove that with probability at least $1/4$, this method yields the r irreducible factors of f in time $O(d\,\mathsf{M}(n)\log q + \mathsf{M}(n)\log(qn)\log r)$.

Solution:

(i) By Lemma 14.6, we have $N(R_i^\times) = \{\gamma \in R_i^\times : \gamma^{q-1} = 1\} = \mathbb{F}_q^\times$, using Fermat's little theorem. By Lagrange's theorem, we have $\#N^{-1}(\gamma) = (q^d-1)/(q-1)$ for all $\gamma \in \mathbb{F}_q^\times$, and hence $N(\gamma)$ is a uniform random element in $\mathbb{F}_q^\times$ if γ is a uniform random element in $R_i^\times$. Similarly, $\chi_i(N(\alpha)) = N(\chi_i(\alpha))$ is a uniform random element of $\mathbb{F}_q^\times$ if α is a uniform random element in $R^\times$.

(ii) The $\chi_i(N(\alpha))$ are independent random variables if $\alpha \in R^\times$ is chosen uniformly at random, by the Chinese Remainder Theorem. Thus for a fixed pair of indices $i < j$, the probability that $\chi_i(N(\alpha))$ and $\chi_j(N(\alpha))$ are equal is $1/(q-1)$. There are $r(r-1)/2 < r^2/2$ such pairs, and hence the probability that all $\chi_i(N(\alpha))$ are distinct is at least $1 - r^2/2(q-1) \geq 1/2$.

(iii) Let $\varphi(t) = (u+t)/(v+t)$ if $t \neq -v$ and $\varphi(-v) = 1$. One verifies that the map ψ with $\psi(t) = (u-vt)/(t-1)$ if $t \neq 1$ and $\psi(1) = -v$ is the inverse of φ, and hence both are bijections. If $t = -v$, then $\pi(u+t) \neq 0 = \pi(v+t)$. Otherwise, we have $\pi(u+t) = \pi(\varphi(t))\pi(v+t)$, since π is multiplicative. Thus $\pi(u+t) \neq \pi(v+t)$ if and only if $t = -v$ or $\pi(\varphi(t)) \neq 1$. Since φ is a bijection, this condition is satisfied for at least $2 + (q-1)/2 > q/2$ elements t of $\mathbb{F}_q$, and the probability estimate follows.

(iv) For a specific choice of $t \in \mathbb{F}_q$, the algorithm succeeds unless either all the $\chi_i(a+t \bmod f)$ are nonzero squares, or they are all nonsquares, or they are all zero. If $a \notin \mathbb{F}_q$, then the last case is impossible and there are two indices $i < j$ with $\chi_i(a \bmod f) \neq \chi_j(a \bmod f)$, by the Chinese Remainder Theorem. Then with probability at least $1/2$ for random t, $\chi_i(a+t \bmod f)$ and $\chi_j(a+t \bmod f)$ are neither both squares nor both nonsquares, by (iii), and the algorithm will separate f_i and f_j.

(v) The algorithm works as follows. Call Algorithm 14.54 with input f and a. If its output is "failure", then call yourself recursively with input f and a (this leads to infinite recursion if $a \in \mathbb{F}_q$). Otherwise, in case of success, call yourself recursively with input g_2 and a rem g_2, and also with input f/g_2 and a rem (f/g_2). Then the bad case $a \in \mathbb{F}_q$ occurs somewhere during the recursive process if and only if initially $\chi_i(a \bmod f)$ and $\chi_j(a \bmod f)$ are the same elements of $\mathbb{F}_q$ for some $i < j$.

Now we assume that this is not the case. Then for a fixed pair of indices $i < j$, the probability that f_i and f_j are not yet separated at depth l of the recursion tree is at most 2^{-l}, by (iv). A similar analysis as in the proof of Theorem 14.11 gives the probability estimate and the time bound.

(vi) By (ii), the probability that $\chi_i(a \bmod f) \neq \chi_j(a \bmod f)$ for all $i < j$ is at least $1/2$, and by (v), the conditional probability that the algorithm is successful in that case is also at least $1/2$, so that the total success probability is at least $(1/2)^2$. The cost for computing a is $O(d\,\mathsf{M}(n)\log q)$ arithmetic operations in $\mathbb{F}_q$, and the time estimate follows from (v).

14.18 The irreducible factors are x^2+x+1 and x^4+x^3+1.

14.21 The ith coefficient of $(ux+v)g$ is vg_0 if $i=0$, $ug_{i-1}+vg_i$ if $0<i<n$, and ug_{n-1} if $i=n$.

(i) We have $|g_0| = |f_0/v| \le A/|v|$. Inductively, we find for $1 \le i < n$ that

$$|g_i| = \left|\frac{f_i - ug_{i-1}}{v}\right| \le \frac{A+|ug_{i-1}|}{|v|} \le \frac{(i+1)A}{|v|},$$

and the last claim follows from $|v| \ge 1$.

(ii) We proceed as in (i), by induction on i. The case $i=0$ is clear. For $1 \le i < n$, we have

$$|g_i| \le \frac{A+|ug_{i-1}|}{|v|} = \frac{A}{|v|} + \alpha|g_{i-1}| \le \frac{A}{|v|}\left(1+\alpha\frac{1-\alpha^i}{1-\alpha}\right) = \frac{A}{|v|}\cdot\frac{1-\alpha^{i+1}}{1-\alpha}.$$

Finally,

$$\frac{1-\alpha^{i+1}}{(1-\alpha)|v|} \le \frac{1}{(1-\alpha)|v|} = \frac{1}{|v|-|u|} \le 1$$

if $|u| < |v|$. If $|u| > |v|$, then we take reversals (Section 9.1) and apply what we just have shown to $\mathrm{rev}_n(f) = (vx+u)\,\mathrm{rev}_{n-1}(g)$.

14.22 (i) We proceed by induction on r. If $r=1$, then $f = f_1^{e_1}$ and $f' = e_1 f_1' f_1^{e_1-1} = e_1 f_1' f/f_1$, by the chain rule. If $r>1$, we write $f = g f_r^{e_r}$, with $g = f_1^{e_1}\cdots f_{r-1}^{e_{r-1}}$. Then

$$(g f_r^{e_r})' = g' f_r^{e_r} + g e_r f_r' f_r^{e_r-1} = \sum_{1\le i<r} e_i f_i' \frac{g f_r^{e_r}}{f_i} + e_r f_r' \frac{g f_r^{e_r}}{f_r} = \sum_{1\le i\le r} e_i f_i' \frac{f}{f_i},$$

by the Leibniz rule and the induction hypothesis.

(ii) Let $1 \le i \le r$. As discussed in the text, the polynomial $f_i^{e_i-1}$ divides f', and $f_i^{e_i}$ divides f' if and only if $e_i f_i' = 0$. Thus f_i^2 does not divide $f/\gcd(f,f')$, and f_i divides it if and only if $e_i f_i' \neq 0$.

14.23 (i) Let $f = x^{1000}+2 \in \mathbb{F}_5[x]$. Then $f' = 0$ and $\gcd(f,f') = f \neq 1$, so that f is not squarefree. More precisely, $f' = 0$ implies that f is a 5th power, namely $f = (x^{200}+2)^5 = (x^8+2)^{5^3}$.

(ii) The claim is false. A counterexample is given by $f = g = x$, where x is the squarefree part of f, of g, and of fg. The correct statement is that the squarefree part of fg is the least common multiple of the squarefree parts of f and g.

14.25 (i) $x^3-3x^2+4 = (x+1)(x-2)^2$ is not squarefree,

(ii) $x^3-2x^2-x+2 = (x+1)(x-1)(x-2)$ is squarefree.

14.27 We should assume that the f_i are monic.

(i) follows from Exercise 14.22 (ii) together with the fact that $f_i' \neq 0$ for all i.

(ii) Since $e_i \le n$ for all i, we have

$$\gcd(u,v^n) = \gcd\left(\prod_{p\nmid e_i} f_i^{e_i-1} \prod_{p\mid e_i} f_i^{e_i}, \prod_{p\nmid e_i} f_i^n\right) = \prod_{p\nmid e_i} f_i^{e_i-1},$$

and the first claim follows. We first calculate v^n rem u with repeated squaring, taking $O(\log n)$ multiplications modulo u or $O(\mathsf{M}(n)\log n)$ field operations. Computing $\gcd(u, v^n \text{ rem } u)$ takes $O(\mathsf{M}(n)\log n)$ field operations as well. Finally, dividing u by the gcd takes only $O(\mathsf{M}(n))$ field operations.

(iii) ALGORITHM 14.57 Squarefree part over finite fields.
Input: A monic polynomial $f \in \mathbb{F}_q[x]$ of degree $n \ge 1$.
Output: The squarefree part of f.

1. $u \longleftarrow \gcd(f,f'), \quad v \longleftarrow \dfrac{f}{u}, \quad w \longleftarrow \dfrac{u}{\gcd(u,v^n)}$
2. call the algorithm recursively to compute the squarefree part z of $w^{1/p}$
3. **return** vz

Let $S(n)$ denote the cost of the algorithm and $m = \deg w$. Steps 1 and 3 take $O(\mathsf{M}(n)\log n)$ field operations. Computing $w^{1/p}$ in step 2 amounts to calculating m/p many (q/p)th powers in $\mathbb{F}_q$, taking at most $2(m/p)\log(q/p)$ operations, where log is the binary logarithm. The degree of z in step 4 is m/p, so that we have the following recursive relation:

$$S(n) \in S\Big(\frac{m}{p}\Big) + O\Big(\mathsf{M}(n)\log n + n\log\frac{q}{p}\Big), \tag{15}$$

and $S(1) = 0$. If c is the implied constant in (15), $d = pc/(p-1)$, and $T(n) = \mathsf{M}(n)\log n + n\log(q/p)$, we claim that $S(n) \le dT(n)$. Indeed, we find inductively

$$S(n) \le S\Big(\frac{m}{p}\Big) + cT(n) \le dT\Big(\frac{m}{p}\Big) + cT(n) \le \Big(\frac{d}{p}+c\Big)T(n) = dT(n),$$

using $p \cdot \mathsf{M}(m/p) \le \mathsf{M}(m)$ and $m \le n$. Thus $S(n) \in O(\mathsf{M}(n)\log n + n\log(q/p))$.

14.30 (i) Lemma 14.22 is valid more generally for perfect fields, so in particular for finite fields. This together with Exercise 14.27 (i) implies the claim.

(ii) Using the same facts as in (i), the invariants

$$h_i = \prod_{j \equiv i \bmod p} g_j \text{ if } i \ge 1, \quad v_{i+1} = \prod_{j \text{ rem } p > i} g_j, \quad w_{i+1} = \sum_{j \text{ rem } p > i} (j-i) g_j' \frac{v_{i+1}}{g_j}$$

are easily proved for $0 \le i \le p$ by induction on i, as in the proof of Theorem 14.23. In particular, this shows that the algorithm stops with $k < p$.

(iii) Replace step 3 by the following steps.

3. $z \longleftarrow \dfrac{f}{h_1 h_2^2 \cdots h_k^k}$
   if $z = 1$ **then return** $(h_1, \ldots, h_k)$
   call the algorithm recursively to compute the squarefree decomposition $(s_1, \ldots, s_l)$ of $z^{1/p}$
4. **for** $i = k+1, \ldots, p-1$ **do** $h_i \longleftarrow 1$
   for $i = 1, \ldots, p-1$ and $j = 1, \ldots, l$ **do** $t_{jp+i} \longleftarrow \gcd(h_i, s_j)$
   for $j = 1, \ldots, l$ **do** $t_{jp} \longleftarrow \dfrac{s_j}{t_{jp+1} t_{jp+2} \cdots t_{(j+1)p-1}}$
   for $i = 1, \ldots, p-1$ **do** $t_i \longleftarrow \dfrac{h_i}{t_{p+i} t_{2p+i} \cdots t_{lp+i}}$
5. let $r < (l+1)p$ be maximal with $t_r \ne 0$
   return $(t_1, \ldots, t_r)$

We note that $s_i = \prod_{jp \le i < (j+1)p} g_i$ for $1 \le j \le l$ at the end of step 3. Thus g_i divides h_j if and only if $j = i \text{ rem } p$, and it divides s_j if and only if $j = \lfloor i/p \rfloor$. This implies the correctness.

The cost for computing z in step 3 is $O(\mathsf{M}(n)\log n)$ field operations, and computing $z^{1/p}$ takes $O((n/p)\log(q/p))$ operations. To compute all gcd's in step 4 efficiently, we proceed as follows. We first compute $s = s_1 \cdots s_l$, of degree at most $\lfloor n/p \rfloor$, taking $O(\mathsf{M}(\lfloor n/p \rfloor)\log n)$ operations. Then we reduce each h_i modulo s, taking $O(\mathsf{M}(\deg h_i))$ field operations, in total $O(\mathsf{M}(n))$ since $\sum_{1 \le i < p} \deg h_i \le n$ and M is superlinear. Then for each i, computing $\gcd(h_i, s_j) = \gcd(h_i \text{ rem } s, s_j)$ for $1 \le j \le l$ takes $O(\mathsf{M}(\lfloor n/p \rfloor)\log n)$ operations, by Exercise 11.4, and $O(\mathsf{M}(n)\log n)$ in total for all i. The cost for the divisions in step 4 is $O(\mathsf{M}(\deg s_j))$ per s_j and $O(\mathsf{M}(\deg h_i))$ per h_i, in total $O(\mathsf{M}(n))$. Thus together with the proof of Theorem 14.23, we find that the cost for all steps except the recursive call in step 3 is $O(\mathsf{M}(n)\log n + n\log(q/p))$, and the running time estimate follows from $\deg z^{1/p} \le n/p$, as in the solution to Exercise 14.27.

14.32 (i) If $f = f_1^{e_1} \cdots f_r^{e_r}$, with all $f_i \in \mathbb{F}_q[x]$ irreducible, monic, and distinct, and positive $e_1, \ldots, e_r$, then $h = \prod_{e_i \text{ odd}} f_i$ yields the desired decomposition. Let

$P_n, S_n \subseteq \mathbb{F}_q[x]$ be the set of all monic and monic squarefree polynomials, respectively, of degree n. We have just shown that P_n is the disjoint union of the sets $P_k^{(2)} \cdot S_{n-2k}$ for $0 \le 2k \le n$, where $P_k^{(2)}$ is the set of all squares of polynomials in P_k. Thus

$$q^n = \#P_n = \sum_{0 \le 2k \le n} \#P_k^{(2)} \cdot \#S_{n-2k} = \sum_{0 \le 2k \le n} q^k s_{n-2k}$$

holds for $n \ge 0$.

(ii) Every monic linear polynomial is squarefree, so that $s_1 = q$, and 1 is the only monic squarefree polynomial of degree zero, whence $s_0 = 1$. Now let $n \ge 2$. Subtracting q times the formula from (i) for $n-2$ from the formula for n, we find

$$\begin{aligned} q^n - q^{n-1} &= \sum_{0 \le 2k \le n} q^k s_{n-2k} - \sum_{0 \le 2k \le n-2} q^{k+1} s_{n-2-2k} \\ &= s_n + \sum_{2 \le 2k \le n} q^k s_{n-2k} - \sum_{2 \le 2(k+1) \le n} q^{k+1} s_{n-2(k+1)} = s_n. \end{aligned}$$

14.33 Let $g \in F[x]$ be a nonconstant irreducible factor of f of degree n. Then $g = (x - a^{1/p})^n$, since $F(a^{1/p})[x]$ is a UFD. The coefficient of x^{n-1} in g is $-na^{1/p}$. This is an element of F, and if $n < p$, then n is a unit in F and $a^{1/p} \in F$, contradicting our assumption that a has no pth root in F. Thus $n = p$ and $f = g$ is irreducible.

14.34 The claim is wrong. If we let $q = 3$, $f = x^2 + 1$, and $\alpha = x + 1 \bmod f$, then we have $\alpha^q = x^3 + 1 \bmod f = -x + 1 \bmod f$, $\xi = x^3 \bmod f = -x \bmod f$, $\check{\xi} = -x$, and $\check{\xi}(\alpha) = -\alpha = -x - 1 \bmod f$.

14.35 By Exercise 10.2, the cost for the ith iteration of step 2 of Algorithm 14.26 is at most

$$\left(2\frac{n}{2^{i-1}} + 1 + \frac{11}{2}(i-1)\right)\mathsf{M}(2^{i-1}) + O\left(\left(\frac{n}{2^{i-1}} + (i-1)\right)2^{i-1}\right) \qquad (16)$$

operations in R, for $1 \le i \le l$, and the cost for step 3 can be bounded by the same estimate with $i = l + 1$. Using $l = \log_2 d$ and the superlinearity of M and summing (16) for $1 \le i \le l + 1$, we find an overall estimate of no more than

$$\left(2\frac{n}{d} + 11\right)\mathsf{M}(d)\log_2 d + 2\mathsf{M}(d) + O(n \log d)$$

operations in R. Thus we may choose $c_1 = 2$ and $c_2 = 11$. Using Exercise 10.9, we can even achieve $c_2 = 7$.

14.39 In step 1 of Algorithm 14.31, we compute x^p rem f, taking $O(\mathsf{M}(n)\log p)$ operations in $\mathbb{F}_q$. Similarly, we compute $a^{(p-1)/2}$ rem f in step 6, at a cost of $O(\mathsf{M}(n)\log p)$. The matrix Q in step 2 is now a $kn \times kn$ matrix over $\mathbb{F}_p$, and Gaussian elimination in step 3 takes $O((nk)^\omega)$ operations in $\mathbb{F}_p$. In step 4, the c_i are chosen from $\mathbb{F}_p$, and each execution of that step takes $O(rnk)$ operations in $\mathbb{F}_p$.

The cost for all other steps it the same as in the original algorithm. Each arithmetic operation in $\mathbb{F}_q$ takes $O(\mathsf{M}(k)\log k)$ operations in $\mathbb{F}_p$. The cost for steps 1 and 3 of the "absolute" algorithm dominates the cost of the other steps, and the overall cost is $O(\mathsf{M}(n)\,\mathsf{M}(k)(\log k)\log p + n^\omega k^\omega)$ or $O^\sim(nk\log p + n^\omega k^\omega)$ arithmetic operations in $\mathbb{F}_p$. In contrast, the original algorithm takes $O(\mathsf{M}(n)k\,\mathsf{M}(k)(\log k)\log p + n^\omega\,\mathsf{M}(k)\log k)$ or $O^\sim(nk^2\log p + n^\omega k)$ operations in $\mathbb{F}_p$. Thus for large n or large k, the original algorithm is faster, while for large p, the "absolute" algorithm is preferable.

14.40 (i) It is clear that $\gcd(f, b-a) \mid f$, for all $a \in \mathbb{F}_p$, and since $b-a$ and $b-a^*$ are coprime for distinct $a, a^* \in \mathbb{F}_p$, we find that the product on the right hand side divides f. Conversely, let $g \in \mathbb{F}_q[x]$ be a monic irreducible factor of f. Then $b \equiv a \bmod g$ for some $a \in \mathbb{F}_p$, and $g \mid \gcd(f, b-a)$. Since all irreducible factors of f are pairwise coprime, we see that f divides the product. Now both polynomials divide each other and are monic, so that they are equal.

(ii) We have $r(a) = 0$ if and only if $\gcd(f, b-a)$ is nonconstant, which in turn is equivalent to the existence of an irreducible factor $g \in \mathbb{F}_q[x]$ of f such that $b \equiv a \bmod g$. If $b \notin \mathbb{F}_p$ and $r(a) = 0$, then $\gcd(f, b-a)$ is a nontrivial factor of f.

(iii) Given a monic nonconstant squarefree polynomial $f \in \mathbb{F}_q[x]$, we compute a basis $b_1 \bmod f, \ldots, b_r \bmod f$ of the absolute Berlekamp algebra $\mathcal{B}$, as in Exercise 14.39. Then we find all roots in $\mathbb{F}_p$ of $\gcd(\mathrm{res}_x(f, b_i - y), x^p - x) \in \mathbb{F}_p[y]$, for $1 \le i \le r$. Finally, we obtain the irreducible factorization of f by successively taking gcd's with $b_i - a$ for all roots $a \in \mathbb{F}_p$ of $\mathrm{res}_x(f, b_i - y)$ and $1 \le i \le r$, as in Exercise 14.38. All steps except the root finding can be done in deterministic polynomial time.

14.42 In the 1999 edition, the text of the exercise contains several typos, and we first give a corrected version of it.

This exercise discusses the easiest case of another factoring method based on linear algebra, due to Niederreiter (see Notes 14.8). Let $p \in \mathbb{N}$ be prime.

(i) Prove that for all rational functions $h \in \mathbb{F}_p(x)$, the $(p-1)$st derivative $h^{(p-1)}$ is a pth power.

(ii) Show that for any nonzero polynomial $f \in \mathbb{F}_p[x]$, the rational function $h = f'/f \in \mathbb{F}_p(x)$ is a solution of the differential equation

$$h^{(p-1)} + h^p = 0. \tag{17}$$

Hint: Prove this first when f is squarefree, using Exercise 9.27 over the splitting field of f, and Wilson's theorem (Exercise 14.1). For the general case, employ the squarefree decomposition of f and Exercise 9.27.

(iii) Prove that if $h = g/f \in \mathbb{F}_p(x)$ satisfies (17), with nonzero coprime $f, g \in \mathbb{F}_p[x]$ and f monic, then $\deg g < \deg f$ and f is squarefree.

(iv) Let f, g be as in (iii) and $\lambda_1, \ldots \lambda_n \in E$ the (distinct) roots of f in a splitting field E of f over $\mathbb{F}_p$. By partial fraction decomposition, there exist $d_1, \ldots, d_n \in E$ such that

$$\frac{g}{f} = \sum_{1 \le i \le n} \frac{d_i}{x - \lambda_i}.$$

Show that $y = d_i/(x - \lambda_i)$ solves (17) for $1 \le i \le n$. (Hint: Uniqueness of partial fraction decomposition). Prove that $d_i = d_k \in \mathbb{F}_p$ if λ_i and λ_k are roots of the same irreducible factor of f, and conclude that

$$\frac{g}{f} = \sum_{1 \le j \le r} c_j \frac{f_j'}{f_j}$$

for some $c_1, \ldots, c_r \in \mathbb{F}_p$, where $f_1, \ldots, f_r$ are the distinct monic irreducible factors of f.

(v) Let $f \in \mathbb{F}_p[x]$ be monic of degree n and

$$\mathcal{N} = \{g \in \mathbb{F}_p[x] : \deg g < n \text{ and } h = \frac{g}{f} \text{ solves (17)}\}.$$

Prove that $f_1' f/f_1, \ldots, f_r' f/f_r$ is a basis of $\mathcal{N}$ as a vector space over $\mathbb{F}_p$ if $f = f_1^{e_1} \cdots f_r^{e_r}$ is the factorization of f into irreducible polynomials.

(vi) Now let f be squarefree and $\mathcal{B} \subseteq \mathbb{F}_p[x]/\langle f \rangle$ the Berlekamp algebra of f. Prove that the map $\varphi : \mathcal{N} \longrightarrow \mathcal{B}$ with $\varphi(g) = g \cdot (f')^{-1} \bmod f$ is a vector space isomorphism. Hint: Consider $\varphi(g) \bmod f_j$ for all j.

(vii) Assume that $p > 2$. Let f as in (vi), $g = \sum_{1 \le j \le r} c_j f_j' f/f_j \in \mathcal{N}$ with all $c_i \in \mathbb{F}_p$, and $S \subseteq \mathbb{F}_p^\times$ the set of squares. Show that

$$\gcd(g^{(p-1)/2} - (f')^{(p-1)/2}, f) = \prod_{c_j \in S} f_j,$$

and conclude that this gcd is nontrivial with probability at least $1/2$ if $c_1, \ldots, c_r$ are chosen uniformly at random in $\mathbb{F}_p$ and $\gcd(f, g) = 1$.

Solution:

(i) We first prove the claim for polynomials. It is clear that $(h^{(p-1)})' = h^{(p)} = 0$ for all $h \in \mathbb{F}_p[x]$, and since $\mathbb{F}_p$ is perfect, h is the pth power of a polynomial. Now let $h = g/f$ for two nonzero polynomials $f, g \in \mathbb{F}_p[x]$. Then the Leibniz rule shows that

$$0 = g^{(p)} = (hf)^{(p)} = \sum_{0 \le i \le p} \binom{p}{i} h^{(i)} f^{(p-i)} = h^{(p)} f + h f^{(p)} = h^{(p)} f,$$

and hence $h^{(p)} = 0$. Furthermore, we see by induction on i that $h^{(i-1)} f^i$ is a polynomial for $i \ge 1$. Now $(h^{(p-1)} f^p)' = h^{(p)} f^p + h^{(p-1)} \cdot p f' f^{p-1} = 0$, so that $h^{(p-1)} f^p = u^p$ for some polynomial $u \in \mathbb{F}_p[x]$, and $h^{(p-1)} = (u/f)^p$ is a pth power.

(ii) We start with $f = x - \lambda$, where λ is in an extension field of $\mathbb{F}_p$. Then

$$\left(\frac{1}{x-\lambda}\right)^{(p-1)} = \frac{(-1)^{p-1}(p-1)!}{(x-\lambda)^p} = -\left(\frac{1}{x-\lambda}\right)^p,$$

by Wilson's theorem (Exercise 14.1). If f is a monic squarefree polynomial, with $f = \prod_{1\le i\le n}(x-\lambda_i)$ for distinct $\lambda_1,\ldots,\lambda_n$ in the splitting field of f, then

$$\frac{f'}{f} = \sum_{1\le i\le n} \frac{1}{x-\lambda_i},$$

by Exercise 9.27 (iv), and the claim follows by linearity from what we have shown above. If f is an arbitrary monic polynomial, with squarefree decomposition $f = \prod_{1\le i\le n} f_i^i$, then Exercise 9.27 (iii) shows that

$$\frac{f'}{f} = \sum_{1\le i\le n} i\frac{f_i'}{f_i},$$

and again the claim follows by linearity from the squarefree case.

(iii) Let $\deg(u/w) = \deg u - \deg w$ for all polynomials u, w. Then $p\deg h = \deg h^p = \deg h^{(p-1)} \le \deg h - p + 1$, which implies that $\deg h \le -1$ or $\deg g < \deg f$. Similarly, if $t \in \mathbb{F}_p[x]$ is irreducible and v_t is the t-adic valuation on $\mathbb{F}_p(x)$, so that $v_t(u) = \max\{i \in \mathbb{N}\cup\{\infty\} : t^i \mid u\}$ for a polynomial $u \in \mathbb{F}_p[x]$ and $v_t(u/w) = v_t(u) - v_t(w)$ for nonzero polynomials $u, w \in \mathbb{F}_p[x]$, then $pv_t(h) = v_t(h^p) = v_t(h^{(p-1)}) \ge v_t(h) - p + 1$, so that $v_t(h) \ge -1$. In particular, if $t \mid f$, then $t \nmid g$ since f and g are coprime, and $-1 \le v_t(h) = v_t(g) - v_t(f) = -v_t(f) \le -1$ implies that $t^2 \nmid f$.

(iv) Since raising to the pth power is an endomorphism of $\mathbb{F}_p(x)$, the partial fraction decomposition of $-h^p$ is

$$-\Big(\frac{g}{f}\Big)^p = \sum_{1\le i\le n} \frac{-d_i^p}{(x-\lambda_i)^p}.$$

On the other hand, we have

$$\Big(\frac{g}{f}\Big)^{(p-1)} = \left(\sum_{1\le i\le n} \frac{d_i}{x-\lambda_i}\right)^{(p-1)} = \sum_{1\le i\le n} \frac{d_i(p-1)!}{(x-\lambda_i)^p} = \sum_{1\le i\le n} \frac{-d_i}{(x-\lambda_i)^p},$$

by Wilson's theorem. Thus $d_i^p = d_i$ for all i, by the uniqueness of partial fraction decomposition, and hence $d_i \in \mathbb{F}_p$. If λ_i and λ_k are conjugate (so that they are roots of the same irreducible factor of f in $\mathbb{F}_p[x]$), then d_i and d_k are conjugate as well, and $d_i, d_k \in \mathbb{F}_p$ implies that $d_i = d_k$. Let f_j be such an irreducible factor, and $d_i = c_j$ for all i with $f_j(\lambda_i) = 0$. Then Exercise 9.27 (iv) implies that

$$\sum_{f_j(\lambda_i)=0} \frac{d_i}{x-\lambda_i} = c_j \sum_{f_j(\lambda_i)=0} \frac{1}{x-\lambda_i} = c_j\frac{f_j'}{f_j},$$

and hence

$$\frac{g}{f} = \sum_{1\le i\le n} \frac{d_i}{x-\lambda_i} = \sum_{1\le j\le r} c_j \frac{f_j'}{f_j}.$$

(v) By (ii), each $f_j' f/f_j$ belongs to $\mathcal{N}$. By (iii), if g is in $\mathcal{N}$, then $f/f_1\cdots f_r$ divides g, so that $g/f = g^*/f_1\cdots f_r$ for $g^* = gf_1\cdots f_r/f \in \mathbb{F}_p[x]$, and (iv) implies that g/f is an $\mathbb{F}_p$-linear combination of the f_j'/f_j, or equivalently, g is an $\mathbb{F}_p$-linear combination of the $f_j' f/f_j$. It remains to show that the $f_j' f/f_j$ are linearly independent. If $\sum_{1\le j\le r} c_j f_j' f/f_j = 0$ in $\mathbb{F}_p[x]$, then $\sum_{1\le j\le r} c_j f_j'/f_j = 0$ in $\mathbb{F}_p(x)$. The rational function on the left hand side has a unique partial fraction decomposition with denominators $f_1,\ldots,f_r$ (Lemma 5.29), but also the partial fraction decomposition with all coefficients equal to zero; hence $c_j = 0$ for all j.

(vi) Since f is squarefree, f' is invertible modulo f, and φ is well defined. It is clear that φ is $\mathbb{F}_p$-linear, and it remains to show that $\varphi(\mathcal{N}) = \mathcal{B}$. By the Leibniz rule, we have

$$f' = (f_1\cdots f_r)' \equiv f_j' f/f_j \bmod f_j.$$

Let $g_j = f_j' f/f_j \in \mathcal{N}$ for $1 \le j \le r$. Then

$$g_j (f')^{-1} \equiv g_j (f_j' f/f_j)^{-1} \equiv 1 \bmod f_j,$$

or equivalently, $\varphi(g_j) \bmod f_j = 1$, and $\varphi(g_j) \bmod f_k = 0$ for $k \ne j$. Thus the image under φ of the basis $f_1' f/f_1,\ldots,f_r' f/f_r$ of $\mathcal{N}$ is a basis of $\mathcal{B}$, and the claim is proved.

(vii) Let $\varphi(g) = g^* \bmod f$ and $(f')^{-1} \equiv s \bmod f$. Then

$$s^{(p-1)/2}(g^{(p-1)/2} - (f')^{(p-1)/2}) \equiv (g^*)^{(p-1)/2} - 1 \bmod f,$$

and since s is coprime to f, we have

$$\gcd(g^{(p-1)/2} - (f')^{(p-1)/2}, f) = \gcd((g^*)^{(p-1)/2} - 1, f).$$

The claims now follow from the discussion preceding Algorithm 14.31 and the facts that φ is an isomorphism of vector spaces and g is coprime to f if and only if g^* is.

14.45 In the statement (iii) of Lemma 14.47, we have to assume that k is a prime not dividing n.

(i) Lemma 14.46 shows that $x^n - 1 = \Phi_1\Phi_n = (x-1)\Phi_n$.

(ii) If $\omega \in \mathbb{C}$ is a primitive nth root of unity, then $-\omega$ is a primitive $2n$th root of unity, by Exercise 8.16 (iii). Conversely, if $\omega \in \mathbb{C}$ is a primitive $2n$th root of unity, then $\omega^n = -1$, and $-\omega$ is a primitive nth root of unity. Thus $\phi_n(x)$ and $\phi_{2n}(-x)$ are monic, squarefree, and have the same roots, so that they are equal.

(iii) We have $\varphi(kn) = \varphi(k)\varphi(n) = (k-1)\varphi(n)$. If $\omega \in \mathbb{C}$ is a primitive knth root of unity, then Exercise 8.13 (iii) shows that ω^k is a primitive nth root of unity. Moreover, if ω is a primitive nth root of unity, then so is ω^k, as in Exercise 8.16 (iii). Thus all roots of $\Phi_{kn}\Phi_n$ are roots of $\Phi_n(x^k)$, and since the former polynomial is squarefree, it divides the latter. Now both polynomials are monic of degree $k\varphi(n)$, and hence they are equal.

(iv) Since every prime divisor of k divides n, we have $\varphi(kn) = k\varphi(n)$. Similarly as in (iii), we find that Φ_{kn} divides $\Phi_n(x^k)$, and since both polynomials are monic of degree $\varphi(nk)$, they are equal.

14.46 (i) The claim is clear if one of m or n is not squarefree. If $m = p_1 \cdots p_r$ and $n = q_1 \cdots q_s$, with distinct primes $p_1, \ldots, p_r, q_1, \ldots, q_s$, then $\mu(nm) = (-1)^{r+s} = (-1)^r(-1)^s = \mu(m)\mu(n)$.

(ii) We write $n = mp^e$ for a prime p not dividing m and $m, e \geq 1$. Then

$$\begin{aligned}\sum_{d|n}\mu(d) &= \sum_{d|m}\sum_{0\leq i\leq e}\mu(dp^i) = \sum_{d|m}\sum_{0\leq i\leq e}\mu(d)\mu(p^i) = \sum_{d|m}(\mu(d)\mu(1)+\mu(d)\mu(p))\\ &= \sum_{d|m}(\mu(d)-\mu(d)) = 0,\end{aligned}$$

by (i).

(iii) We have

$$\sum_{d|n}\mu(d)f\Big(\frac{n}{d}\Big) = \sum_{d|n}\mu(d)\sum_{e|(n/d)}g(e) = \sum_{de|n}\mu(d)g(e) = \sum_{e|n}g(e)\sum_{d|(n/e)}\mu(d) = g(n)$$

since the last inner sum vanishes unless $e = n$, by (ii).

(iv) The corresponding formula is

$$g(n) = \prod_{d|n} f(d)^{\mu(n/d)} = \prod_{d|n} f\Big(\frac{n}{d}\Big)^{\mu(d)} \text{ for } n \in \mathbb{N}_{>0},$$

and follows from (iii) by taking logarithms.

(v) This follows from $\sum_{e|n} 1 = d(n)$ by Möbius inversion.

14.47 (i) From $qS_1 = S_1$ and the fact that for all $i, j \in S_1$, $i \equiv j \bmod n$ implies $x^i \equiv x^j \bmod x^n - 1$, it follows that $b_1^q \equiv b_1 \bmod x^n - 1$, and $b_1 \bmod x^n - 1$ belongs to the Berlekamp algebra $\mathcal{B}$. Similarly, $b_2 \bmod x^n - 1, \ldots, b_r \bmod x^n - 1 \in \mathcal{B}$. Assume that $\sum_{1\leq i\leq r}\lambda_i b_i \equiv 0 \bmod x^n - 1$ for some $\lambda_1, \ldots, \lambda_r \in \mathbb{F}_q$. The sum on the left hand side has degree less than n, whence $\lambda_1 = \cdots = \lambda_r = 0$ and the $b_i \bmod x^n - 1$ are linearly independent for $1 \leq i \leq r$.

It remains to show that $b_1 \bmod x^n - 1, \ldots, b_r \bmod x^n - 1$ generate $\mathcal{B}$. Let $f = \sum_{0\leq j<n} f_j x^j \in \mathbb{F}_q[x]$. A similar argument as above shows that for each equivalence class S_i, raising f to the qth power and reducing modulo $x^n - 1$ permutes the coefficients f_j with $j \in S_i$ cyclically. Thus $f \bmod x^n - 1$ is in the Berlekamp algebra $\mathcal{B}$

if and only if for $1 \le i \le r$, all coefficients f_j with $j \in S_i$ are equal, or equivalently, f is a linear combination of $b_1, \ldots, b_r$. In particular, $\mathcal{B}$ is an r-dimensional vector space over $\mathbb{F}_q$.

(ii) We first determine all equivalence classes $S_1, \ldots, S_r$. This takes at most n multiplications by q modulo n. Computing q rem n takes $O((\log q)\log n)$ word operations with classical arithmetic, and the cost for all modular multiplications is $O(n\mathsf{M}(\log n))$ word operations with fast arithmetic. Then we set up $b_1, \ldots, b_r$, repeatedly and independently perform essentially steps 4 through 7 of Berlekamp's algorithm 14.31 and refine the partial factorizations that we obtain, as described in Exercise 14.38 (ii), until we have found a factorization into r factors. The cost is $O(n)$ arithmetic operations in $\mathbb{F}_q$ for step 4 (no additions have to be performed), $O(\mathsf{M}(n)\log n)$ for the modified steps 5 and 7, by Exercise 11.4, and $O(\mathsf{M}(n)\log q)$ for step 6. A similar analysis as in the proof of Theorem 14.11 shows that the expected number of iterations is $O(\log r)$, and hence the expected cost for the second part is $O(\mathsf{M}(n)\log(qn)\log r)$ arithmetic operations in $\mathbb{F}_q$.

Chapter 15

15.3 f is either irreducible, or it splits into one irreducible factor of degree 5 and one of degree 3, or into a linear factor and an irreducible factor of degree 7.

15.4 We have $f = x^4 - 2x^2 + 9$, $f \equiv (x+1)^4 \bmod 2$, $f \equiv x^2(x^2+1) \bmod 3$, and $f \equiv (x^2 - x + 2)(x^2 + x + 2) \bmod 5$. None of the divisors $\pm 1, \pm 3, \pm 9$ of $f(0)$ is a root of f, so that f has no linear factor, and comparing coefficients in the ansatz $(x^2 + ax + b)(x^2 + cx + d) = f$ proves that f has no quadratic factor.

15.7 If $p = p_1$, then $\Phi_n \equiv \Phi_{p_2}^p \bmod p$, and similarly for $p = p_2$. Now we assume that p is a prime different from p_1 and p_2. Then Lemma 14.50 shows that Φ_n splits into $\varphi(n)/d$ irreducible factors of degree $d = \operatorname{ord}_n(p)$. The Chinese Remainder Theorem implies that $d = \operatorname{lcm}(\operatorname{ord}_{p_1}(p), \operatorname{ord}_{p_2}(p)) \mid \operatorname{lcm}(p_1 - 1, p_2 - 1)$. Since $p_1 - 1$ and $p_2 - 1$ are both even, we have $d < \varphi(n) = (p_1 - 1)(p_2 - 1) = \deg \Phi_n$ and $\varphi(n)/d \ge 2$. If $p_1 - 1$ even divides $p_2 - 1$, then $d \mid p_2 - 1$ and $\varphi(n)/d \ge p_1 - 1$.

15.8 If p divides the discriminant of the polynomial, then it may happen that the polynomial has some linear and some quadratic irreducible factors modulo p, as for f_5 below. So we assume that p does not divide the discriminant.

Let $i \in \mathbb{N}$ be positive, $p_1, \ldots, p_i \in \mathbb{N}$ be the first i primes, and $\sqrt{p_j} \in \mathbb{R}$ the positive square root of p_j, for all j. For $e = (e_1, \ldots, e_i) \in \{0,1\}^i$, we write $s_e = (-1)^{e_1}\sqrt{p_1} + \cdots + (-1)^{e_i}\sqrt{p_i}$ for short, so that

$$f = \prod_{e \in \{0,1\}^i} (x - s_e)$$

is the ith Swinnerton-Dyer polynomial. Moreover, we let $p \in \mathbb{N}$ be a prime not dividing the discriminant $\operatorname{res}(f, f')$ of f, so that $f \bmod p$ is squarefree. The field

$\mathbb{F}_{p^2}$ contains the square roots of all p_j, so that f splits into linear factors over $\mathbb{F}_{p^2}$, and all irreducible factors of f mod p in $\mathbb{F}_p[x]$ are at most quadratic; this is also true when p divides the discriminant. We may assume that the p_j are ordered in such a way that $p_1,\ldots,p_t$ are squares and $p_{t+1},\ldots,p_i$ are nonsquares modulo p, for some $t \in \{0,\ldots,i\}$. For $1 \le j \le i$, we let $r_j \in \mathbb{F}_{p^2}$ be a fixed square root of p_j modulo p. Then $r_j \in \mathbb{F}_p$ if and only if $j \le t$.

We let $R = \mathbb{Z}[\sqrt{p_1},\ldots,\sqrt{p_i}]$, and consider the ring homomorphism $\varphi: R \longrightarrow \mathbb{F}_{p^2}$ which maps an integer z to z mod p and $\sqrt{p_i}$ to r_i for all i. Such a homomorphism exists by induction on i: for $i = 0$ this is just the canonical residue class map $\mathbb{Z} \longrightarrow \mathbb{F}_p$. So let us assume that $i \ge 1$ and we already have shown that for $S = \mathbb{Z}[\sqrt{p_1},\ldots,\sqrt{p_{i-1}}]$, there exists a ring homomorphism $\psi: S \longrightarrow \mathbb{F}_{p^2}$ with $\psi(z) = z$ mod p for all $z \in \mathbb{Z}$ and $\psi(\sqrt{p_j}) = r_j$ for $1 \le j < i$. We can extend ψ to the polynomial ring $S[y]$ by mapping y to r_i, and denote the extension also by ψ. Then $\langle y^2 - p_i\rangle \subseteq \ker\psi$, and this implies that the map $\varphi: R = S[\sqrt{p_i}] \cong S[y]/\langle y^2 - p_i\rangle \longrightarrow \mathbb{F}_{p^2}$ with $\varphi(a + b\sqrt{p_i}) = \psi(a) + \psi(b)r_i$ is a well-defined ring homomorphism.

This homomorphism φ can be extended to a ring homomorphism $R[x] \longrightarrow \mathbb{F}_{p^2}[x]$ in a canonical way, by applying φ to each coefficient, and we denote the latter homomorphism by φ as well. Then

$$\varphi(s_e) = \sum_{1\le j\le i} (-1)^{e_j} r_j \text{ for all } e \in \{0,1\}^i,$$

and

$$\varphi(f) = \prod_{e\in\{0,1\}^i} (x - \varphi(s_e)).$$

If all r_j are in $\mathbb{F}_p$, then f splits into these linear factors modulo p.

Since $\varphi(f) = f$ mod p is squarefree, we have $\varphi(s_e) \ne \varphi(s_{e^*})$ for all distinct $e, e^* \in \{0,1\}^n$, and in particular $p \notin \{p_1,\ldots,p_i\}$.

Assume that f does not split into linear factors modulo p. Then $t < i$. Let $g \in \mathbb{F}_p[x]$ be an irreducible factor of f mod p, and $e \in \{0,1\}^i$ be such that $x - \varphi(s_e)$ divides g in $\mathbb{F}_{p^2}$. Then

$$\varphi(s_e)^p = \sum_{1\le j\le t} (-1)^{e_j} r_j^p + \sum_{t<j\le i} (-1)^{e_j} r_j^p = \sum_{1\le j\le t} (-1)^{e_j} r_j - \sum_{t<j\le i} (-1)^{e_j} r_j = \varphi(s_{e^*})$$

holds in $\mathbb{F}_{p^2}$, where $e^* = (e_1,\ldots,e_t,1-e_{t+1},\ldots,1-e_i)$. The Frobenius automorphism $\alpha \longmapsto \alpha^p$ of $\mathbb{F}_{p^2}$ over $\mathbb{F}_p$ permutes the roots of $g \in \mathbb{F}_p[x]$, and hence $\varphi(s_{e^*}) = \varphi(s_e)^p$ is also a root of g in $\mathbb{F}_{p^2}$. Since $e \ne e^*$ and $\varphi(f)$ is squarefree, we conclude that $\varphi(s_e)$ and $\varphi(s_e^*)$ are distinct elements of $\mathbb{F}_{p^2}$, and $\deg g = 2$.

In fact, the discriminant of f may be divisible by primes other than $p_1,\ldots,p_i$. For example, let f_5 be the polynomial corresponding to $i = 5$. Then $13 \mid \operatorname{res}(f_5, f_5')$, so that f_5 mod 13 is not squarefree, but $13 \notin \{p_1,\ldots,p_5\}$. Moreover, f_5 mod 13 has both linear and quadratic irreducible factors.

15.11 $a_0 = 0$, $a_1 = 65$, $a_2 = 1625$.

15.12 Let $f = mf^*$ for a polynomial $f^* \in R[x]$. Then (i) implies that $f^* = q^*g + r^*$ for some $q^*, r^* \in R[x]$ with $\deg r^* < \deg g$. Now $f = mf^* = (mq^*)g + mr^*$ and $f = qg + r$ are both divisions with remainder, and the uniqueness statement of (i) implies that $q = mq^*$ and $r = mr^*$.

15.13 (i) By the uniqueness of Hensel lifting, f factors modulo p^{100} into three monic irreducible and pairwise coprime polynomials of degrees $1, 2$, and 5.

(ii) The possible factorization patterns of f in $\mathbb{Q}[x]$ are $(1,2,5)$, $(3,5)$, $(2,6)$, $(1,7)$, and (8).

(iii) It follows that f is irreducible.

15.17 We have

$$\begin{aligned} s^*g^* + t^*h^* - 1 &\equiv (s - sb - ch^*)g^* + (t - tb - cg^*)h^* - 1 \\ &= sg^* + th^* - 1 - (sg^* + th^*)b \equiv -b^2 \equiv 0 \bmod m^2, \end{aligned}$$

since $b \equiv 0 \bmod m$, by assumption. Moreover, Lemma 15.9 implies that $c \equiv d \equiv 0 \bmod m$, and hence $s^* \equiv s \bmod m$ and $t^* \equiv t \bmod m$. Since $\deg d < \deg h^*$, we have $\deg s^* < \deg h^*$, and $s^*g^* + t^*h^* \equiv 1 \bmod m^2$ together with the fact that h^* is monic implies that $\deg t^* \leq \deg s^* + \deg g^* - \deg h^* < \deg g^*$.

15.18 (i) $u \equiv q^2 \bmod p$ is a unit modulo p, and by symmetry, u is also a unit modulo q. Thus, by the Chinese Remainder Theorem, it is a unit modulo pq.

(ii) $(px+q)(qx+p) = pqx^2 + (p^2+q^2)x + pq \equiv ux \bmod pq$.

(iii) Let $g, h \in \mathbb{Z}[x]$ with $px + q \equiv gh \bmod pq$. Then $q \equiv gh \bmod p$ implies that both g and h are units modulo p, and $px \equiv gh \bmod q$ shows that exactly one of g and h is a unit modulo q, where we use unique factorization of polynomials modulo p and modulo q. Thus, by the Chinese Remainder Theorem, exactly one of g and h is a unit modulo pq, and $px + q$ is irreducible modulo pq.

15.21 (i) We have $\varphi_i = \sum_{j+k=i} g_j h_k - f_i$, and this implies that $\partial\varphi_i/\partial g_j = h_{i-j}$ and $\partial\varphi_i/\partial h_j = g_{i-j}$, for all j, where we let g_j and h_j be zero if the index j is "out of range".

(ii) The fact that $\operatorname{lc}(f)$ is a unit modulo p implies that the leading coefficients of g, h are units modulo p. Thus the Sylvester matrix of $g \bmod p$ and $h \bmod p$ equals the Sylvester matrix of g and h, taken modulo p. Then Exercise 6.15 (ii), which is valid more generally for polynomials with invertible leading coefficients, proves that $s, t \in R[x]$ with $\deg s < \deg h$, $\deg t < \deg g$, and $sf + tg \equiv 1 \bmod p$ exist if and only if $\operatorname{res}(g,h) = \det J$ is a unit modulo p, or equivalently, J is invertible modulo p. Finally, if we have arbitrary s^*, t^* with $s^*g + t^*h \equiv 1 \bmod p$, then we perform one division with remainder $s^* \equiv qh + s \bmod p$, with $\deg s < \deg h$, set $t = t^* + qg$, and then s, t satisfy $sf + tg \equiv 1 \bmod p$ plus the above degree conditions.

(In the 1999 edition, p was called m.)

15.24 Since $\gcd(f, \partial f/\partial x) = 1$ in $F(y)[x]$, the resultant $r = \operatorname{res}_x(f, \partial f/\partial x) \in F[y]$ is a nonzero polynomial of degree less than $2nd$, by Corollary 6.17 and Theorem 6.22. Since $b \mid r$, by Exercise 6.41, the condition in step 2 is satisfied if and only if $r(u) \neq 0$. Now r has less than $2nd$ roots, and the success probability for step 2 is at least $1/2$.

We first show that the condition in step 9 is satisfied if and only if $g^* h^* = b f^*$. The "if" part is clear, and we assume conversely that $\deg_y(g^* h^*) = \deg_y(b f^*)$. By construction, we have $g^* h^* \equiv b f^* \bmod (y-u)^l$. Now both sides have degree $\deg_y b + \deg_y f^* < l$, and hence they are equal.

For a polynomial $v \in F[x,y]$, we denote by $\mu(v)$ the number of irreducible factors in $F[x]$ of $v(x,u)$, and show the invariants

$$\begin{aligned} &f^* \equiv b \textstyle\prod_{i\in T} g_i \bmod (y-u)^l, \quad b = \operatorname{lc}_x(f^*), \quad f = f^* \textstyle\prod_{g\in G} g, \\ &\text{each polynomial in } G \text{ is irreducible,} \\ &f^* \text{ is primitive with respect to } x \text{ and each of its irreducible factors} \\ &v \in F[x,y] \text{ has } \mu(v) \geq s \end{aligned} \tag{6}$$

of the loop 6 by induction. This is clear initially, and we assume that the conditions hold before step 8, and that the condition in step 9 is true for some subset $S \subseteq T$ of cardinality s. As in the proof of Theorem 15.3, we then find that the invariants hold again at the next pass through step 6. Now we assume that the condition in step 9 is false for all subsets S, but that f^* has an irreducible factor $g \in F[x,y]$ with $\mu(g) = s$. Let $h = f^*/g$. Since $F[x]$ is a UFD, there is a subset $S \subseteq T$ with $\#S = s$, $g \equiv \operatorname{lc}_x(g) \prod_{i\in S} h_i \bmod y-u$, and $h \equiv \operatorname{lc}_x(h) \prod_{i\in T\setminus S} h_i \bmod y-u$. As in the proof of Theorem 15.20, the uniqueness of Hensel lifting implies that $g \equiv \operatorname{lc}_x(g) \prod_{i\in S} g_i \bmod (y-u)^l$, $h \equiv \operatorname{lc}_x(h) \prod_{i\in T\setminus S} g_i \bmod (y-u)^l$, and $g^* \equiv \operatorname{lc}_x(h) g \bmod (y-u)^l$ and $h^* \equiv \operatorname{lc}_x(g) h \bmod (y-u)^l$ in step 8, so that the condition in step 9 is satisfied for the particular subset S. This contradiction proves that $\mu(f^*) \geq s+1$, and the invariants hold again after step 10. Finally, as in the proof of Theorem 15.3, f^* is irreducible if $2s > \#T$ in step 6, which together with (6) proves that the algorithm returns the correct result.

The cost for evaluating f at $y = u$ in step 2 is $O(nd)$ arithmetic operations in F, and the gcd takes $O(\mathsf{M}(n)\log n)$ field operations. By the above, the expected cost for step 2 is $O(nd + \mathsf{M}(n)\log n)$ field operations. The estimate for step 3 in the finite field case is from Corollary 14.30, and Theorem 15.18 gives the cost for step 4. The cost for computing g^* and h^* in step 8 is $O(\mathsf{M}(n)\log n)$ additions and multiplications on polynomials in $F[y]$ of degree at most d, or $O(\mathsf{M}(n)\log n \cdot \mathsf{M}(d))$ field operations, by Corollary 10.8, and computing the primitive parts in step 9 takes $O(n\mathsf{M}(d)\log d)$ field operations. The number of iterations is determined as in the proof of Theorem 15.3.

15.25 (i) Since g is primitive with respect to both x and y, Theorem 14.20 implies that $u = v$ is the squarefree part of g. Thus $V \cup W$ contains all irreducible factors of g after step 4, and the algorithm returns the correct result in step 7.

(ii) If h is a pth power, then clearly $\partial h/\partial x = \partial h/\partial y = 0$. Conversely, if both partial derivatives vanish, then h is a polynomial in x^p and y^p, and the claim follows since $\mathbb{F}_q$ is a perfect field.

(iii) We write $h_x = \partial h/\partial x$ for short, and assume that $\gcd(h, h_x)$ is constant in $\mathbb{F}_q[x,y]$. In particular, this implies that h is primitive with respect to x. Exercise 14.22 (ii) shows that an irreducible factor $w \in \mathbb{F}_q[x,y]$ of h with multiplicity e divides h_x if and only if $e > 1$ or $w_x = 0$. Since $\gcd(h, h_x)$ is constant, we have $e = 1$ (and $w_x \neq 0$).

If h is squarefree, then $\gcd(h, h_x)$ is the product of all irreducible divisors $w \in F[x,y]$ of h with $w_x = 0$. In characteristic zero, no such divisors exist, and the reverse statement is true. However, in positive characteristic it may fail to hold. For example, the polynomial $h = (y - x^p)(x - y^p) \in \mathbb{F}_q[x,y]$ is squarefree, but $\gcd(h, h_x) = y - x^p$ and $\gcd(h, h_y) = x - y^p$.

(iv) Since g is primitive with respect to x, Exercise 14.22 (ii) shows that u is squarefree and every irreducible factor $h \in \mathbb{F}_q[x,y]$ of g has $h_x \neq 0$. Again by the same Exercise, we find that $\gcd(u, u_x) = 1$. The other claims follow from $w \mid u$ and analogous arguments.

(v) Since h is irreducible, (ii) implies that one of h_x and h_y is nonzero. If $p \nmid e$, then Exercise 14.22 (ii) shows that $h \mid u$ or $h \mid v$, and hence $h \mid vw$. Then $h \in V \cup W$, and h^e is removed from g in step 5. On the other hand, if $p \mid e$, then h divides neither u nor v, and h^e divides g after step 5.

(vi) By (v), g is a pth power in step 6.

15.26 In the 1999 edition, the definition in part (ii) of this exercise must be changed so that only h_k is required to be nonconstant, some of $h_1, \ldots, h_{k-1}$ may well be constant.

(i) Let K be the field of fractions or R and $g_1^*, \ldots, g_m^* \in K[x]$ be the polynomials in the monic squarefree decomposition of $f/\mathrm{lc}(f)$ in $K[x]$, and let $g_1, \ldots, g_m \in R[x]$ be primitive scalar multiples of $g_1^*, \ldots, g_m^*$. Then the g_i are squarefree and pairwise coprime in $K[x]$, since the g_i^* are. Now $f = \mathrm{lc}(f) \prod_{1 \le i \le m} (g_i^*)^i$, together with Gauß' lemma, implies that $g = \prod_{1 \le i \le m} g_i^i \in \mathbb{Z}[x]$ is a primitive polynomial and $f = cg$ for a unit $c \in R^\times$. Multiplying g_m by c if necessary, we may assume that $c = 1$. This proves the existence of $g_1, \ldots, g_m$. The uniqueness up to multiplication by units in $R^\times$ follows from the uniqueness of the monic squarefree decomposition in $K[x]$.

(In the 1999 edition, we have $R = \mathbb{Z}$ and $K = \mathbb{Q}$, and f and the g_i have positive leading coefficients, so that $c = 1$.)

(ii) The goal of this part is to show that the modular image of the primitive squarefree decomposition of f is equal to the squarefree decomposition of the modular image of f, for all primes except those dividing $\mathrm{res}(g, g')$.

Each irreducible factor of f in $\mathbb{Z}[x]$ divides the squarefree part $g = g_1 \cdots g_m$ of f in $\mathbb{Z}[x]$, and hence each irreducible factor of f mod p divides g mod p. In particular, the squarefree part $(h_1 \cdots h_k) \bmod p \in \mathbb{F}_p[x]$ divides g mod p. Let $1 \le$

$i \le m$, let $g^* \in \mathbb{Z}[x]$ be an irreducible factor of g_i, and $h \in \mathbb{Z}[x]$ be monic such that $h \bmod p$ is an irreducible factor of $g^* \bmod p$. Then h^i divides f modulo p, so that the multiplicity of $h \bmod p$ in $f \bmod p$ is at least i, and hence h divides $h_i \cdots h_k$ modulo p and does not divide $h_1 \cdots h_{i-1}$ modulo p. In particular, taking $i = m$ implies that $k \ge m$.

Now we assume that p does not divide $\operatorname{res}(g, g')$. Since p does not divide $\operatorname{lc}(f)$, it does not divide $\operatorname{lc}(g)$ either, and the discriminant of $g \bmod p$ is equal to $\operatorname{res}(g, g') \bmod p \neq 0$. Thus $g \bmod p$ is a squarefree divisor of $f \bmod p$ which is divisible by the squarefree part of $f \bmod p$, by the above, and hence

$$g_1 \cdots g_m = g \equiv \operatorname{lc}(g) h_1 \cdots h_k \bmod p. \tag{7}$$

Since $g \bmod p$ is squarefree, the $g_i \bmod p$ are squarefree and pairwise coprime. We have seen above that any irreducible factor of $(g_2 \cdots g_m) \bmod p$ is coprime to $h_1 \bmod p$ and divides $(h_2 \cdots h_k) \bmod p$. Thus $g_2 \cdots g_m$ is coprime to h_1 modulo p and divides $h_2 \cdots h_k$ modulo p, and hence h_1 divides g_1 modulo p. Assume that $g_1 \bmod p$ and $h_2 \cdots h_k \bmod p$ have a monic irreducible common divisor $h \bmod p$, for some $h \in \mathbb{Z}[x]$. Then $h \bmod p$ divides $g_1 \bmod p$ exactly once and $f \bmod p$ at least twice, so that it divides $(f/g_1) \bmod p$. Since $h \bmod p$ is irreducible, it divides $g_i \bmod p$ for some $i \ge 2$. But the latter polynomial is coprime to $g_1 \bmod p$, and this contradiction proves that g_1 and $h_2 \cdots h_k$ are coprime modulo p and $g_1 \equiv \operatorname{lc}(g_1) h_1 \bmod p$. Dividing both sides in 7 by g_1 and proceeding inductively, we find that $g_i \equiv \operatorname{lc}(g_i) h_i \bmod p$ for $1 \le i \le m$, and this also implies that $k = m$.

We note that $k = m$ does not imply that $g \bmod p$ is squarefree: a counterexample is given by $f = x^4 + x^2 = (x^2+1) \cdot x^2 \equiv (x^2+x)^2 \bmod 2$.

(iv) Here is the algorithm.

ALGORITHM 15.25 Small primes modular squarefree decomposition.

Input: A nonconstant primitive polynomial $f \in \mathbb{Z}[x]$ of degree n and max-norm $\|f\|_\infty = A$ and with $\operatorname{lc}(f) > 0$.

Output: The primitive squarefree decomposition of f in $\mathbb{Z}[x]$.

1. $b \longleftarrow \operatorname{lc}(f), \quad B \longleftarrow (n+1)^{1/2} 2^n A b$
   $k \longleftarrow \lceil 2 \log_2(n^n B^{2n-1}) \rceil, \quad l \longleftarrow \lceil \log_2(2B) \rceil$
2. **repeat**
3. choose a set S_0 of $2l$ odd primes, each less than $2k \ln k$
4. $S_1 \longleftarrow \{p \in S_0 : p \nmid b\}$
   for each $p \in S_1$ **call** Yun's algorithm in $\mathbb{F}_p[x]$ (Exercise 14.30) to compute the monic squarefree decomposition $f \equiv b \prod_{1 \le i \le m_p} h_{p,i}^i \bmod p$, with all $h_{p,i} \in \mathbb{Z}[x]$ monic and with coefficients in $\{0, \ldots, p-1\}$
5. $e \longleftarrow \max\{\deg(h_{p,1} \cdots h_{p,m_p}) : p \in S_1\}, \quad s \longleftarrow \min\{m_p : p \in S_1\}$
   $S_2 \longleftarrow \{p \in S_1 : \deg(h_{p,1} \cdots h_{p,m_p}) = e\}$
   if $\#S_2 \ge l$ **then** remove $\#S_2 - l$ elements from S_2 **else goto** 3

6. **for** $i = 1, \ldots, s$ **call** the Chinese Remainder Algorithm 5.4 to compute $g_i^* \in \mathbb{Z}[x]$ with max-norm less than $(\prod_{p \in S_2} p)/2$ and $g_i^* \equiv b h_{p,i} \bmod p$ for all $p \in S_2$
7. **until** $\prod_{1 \le i \le s} \operatorname{lc}(\operatorname{pp}(g_i^*))^i = b$ and $\prod_{1 \le i \le s} \| \operatorname{pp}(g_i^*) \|_1^i \le B$
8. **return** $\operatorname{pp}(g_1^*), \ldots, \operatorname{pp}(g_s^*)$

The proof of correctness and timing estimates are in Gerhard (2001a).

In practice, the same remarks as for the modular gcd algorithm 6.38 apply. Instead of choosing $2l$ primes, one would work adaptively by starting with only l or even fewer, check whether the constant coefficients of f and $\prod_{1 \le i \le s} \operatorname{pp}(g_i^*)^i$ agree in step 7, and if not, add some more primes dynamically. If the constant coefficients agree, then one would check whether in fact $f = \prod_{1 \le i \le m} \operatorname{pp}(g_i^*)^i$ holds. Concerning the size of the primes, it is advantageous to choose single precision primes fitting precisely into one machine word, maybe even deterministically from a precomputed list, instead of the first k primes. For example, if the word size of our processor is 64, then $l = \lceil \log_2(2B)/63 \rceil$ primes between 2^{63} and 2^{64} are sufficient to reconstruct the gcd.

15.27 Here is the algorithm.

ALGORITHM 15.26 Prime power modular squarefree decomposition.

Input: A nonconstant primitive polynomial $f \in \mathbb{Z}[x]$ of degree n and max-norm $\| f \|_\infty = A$ and with $\operatorname{lc}(f) > 0$.

Output: The primitive squarefree decomposition of f in $\mathbb{Z}[x]$, as defined in Exercise 15.26.

1. **call** the modular gcd algorithm 6.38 to compute
   $u \longleftarrow \gcd(f, f')$, $\quad v_1 \longleftarrow \dfrac{f}{u}$, $\quad w_1 \longleftarrow \dfrac{f'}{u}$
2. $b \longleftarrow \operatorname{lc}(v_1)$, $\quad B \longleftarrow (n+1)^{1/2} 2^n A b$
   $k \longleftarrow \lceil 2 \log_2(n^n B^{2n-1}) \rceil$
3. choose an odd prime p with $n < p \le 2k \ln k$ such that $p \nmid b$ and $v_1 \bmod p$ is squarefree
   $l \longleftarrow \lceil \log_p(2B) \rceil$
4. $i \longleftarrow 1$
   repeat
   compute the polynomials $h_i, v_{i+1}, w_{i+1} \in \mathbb{Z}[x]$ with coefficients in the set $\{0, \ldots, p-1\}$ such that $h_i \bmod p = \gcd(v_i \bmod p, w_i - v_i' \bmod p)$, h_i is monic, $v_i \equiv h_i v_{i+1} \bmod p$, and $w_i - v_i' \equiv h_i w_{i+1} \bmod p$
   $i \longleftarrow i + 1$
   until $\deg v_i = 0$
   $s \longleftarrow i - 1$

5. **call** the multifactor Hensel lifting algorithm 15.17 to compute a factorization $v_1 \equiv b\prod_{1\le i\le s} h_i^*$, with all $h_i^* \in \mathbb{Z}[x]$ monic of max-norm less than $p^l/2$ such that $h_i^* \equiv h_i \bmod p$
6. **for** $i = 1,\ldots,s$ compute $g_i^* \in \mathbb{Z}[x]$ of max-norm less than $p^l/2$ such that $g_i^* \equiv bh_i^* \bmod p^l$
7. **return** $\operatorname{pp}(g_1^*),\ldots,\operatorname{pp}(g_s^*)$

This variant of Algorithm 14.21 appears essentially in Yun (1976). Let $f = \prod_{1\le i\le m} g_i^i$ be the primitive squarefree decomposition of f in $\mathbb{Z}[x]$, with primitive squarefree and pairwise coprime $g_i \in \mathbb{Z}[x]$ with positive leading coefficients. Then $v_1 = g_1\cdots g_m$. Now assume that p satisfies the conditions in step 3, or equivalently, that it does not divide $\operatorname{res}(v_1, v_1')$. By Exercises 14.30 and 15.26 (ii), we have $s = m$ and $(b/\operatorname{lc}(g_i))g_i \equiv bh_i \bmod p$ for $1 \le i \le m$ and $v_1 \equiv b\prod_{1\le i\le m} h_i \bmod p$. The uniqueness of Hensel lifting (Theorem 15.14) implies that $(b/\operatorname{lc}(g_i))g_i \equiv bh_i^* \equiv g_i^* \bmod p^l$. Now both sides have max-norms less than $p^l/2$, by Mignotte's bound 6.33 and the choice of l, and hence they are equal. Since both g_i and $\operatorname{pp}(g_i^*)$ are primitive and have positive leading coefficients, we find that $g_i = \operatorname{pp}(g_i^*)$. This shows that the algorithm works correctly.

Step 1 takes $O^\sim(n^2 + n\log A)$ word operations, by Corollary 11.11. We have $\|v_1\|_\infty, \|w_1\|_\infty \le nB$, by Mignotte's bound 6.33. The cost for one execution of step 3 is $O(n\log B\cdot\log k)$ word operations for reducing all coefficients of v_1 and w_1 modulo p and $O(\mathsf{M}(n)\log n\cdot\mathsf{M}(\log k) + n\mathsf{M}(\log k)\operatorname{loglog} k)$ word operations for computing $\gcd(v_1 \bmod p, v_1' \bmod p)$ to check whether $v_1 \bmod p$ is squarefree. Now p divides b if and only if it divides $\operatorname{lc}(v_1)$, and the latter in turn divides $\operatorname{res}(v_1, v_1')$, by Exercise 6.41. Thus $p \nmid b$ and $\gcd(v_1 \bmod p, v_1' \bmod p) = 1$ for a prime p if and only if $p \nmid \operatorname{res}(v_1, v_1')$. Since $\|v_1\|_2 \le B$ and $\|v_1'\|_2 \le nB$, Theorem 6.23 implies that $|\operatorname{res}(v_1, v_1')| \le n^n B^{2n-1} \le 2^{k/2}$. Since g is squarefree, its discriminant is nonzero and has at most $k/2$ prime divisors, and if we choose p uniformly at random from among the first k primes exceeding n, then the expected number of iterations of step 3 is at most 2. (We may even allow primes smaller than n if we modify step 4 according to Exercise 14.30). We ignore the cost for prime finding. Step 4 takes $O(\mathsf{M}(n)\log n\cdot\mathsf{M}(\log k)\operatorname{loglog} k)$ word operations, by Exercise 14.30. The cost for step 5 is

$$O\Big((\mathsf{M}(n)\mathsf{M}(l\log k) + \mathsf{M}(n)\log n\cdot\mathsf{M}(\log k) + n\mathsf{M}(\log k)\operatorname{loglog} k)\log n\Big)$$

word operations, by Theorem 15.18. Steps 6 and 7 take $O(n\mathsf{M}(l\log k)\log(l\log k))$ word operations. Using $k \in O(n^2 + n\log A)$, $\log k \in O(\log(n\log A))$, and $l\log k \in O(n + \log A)$, the overall cost for steps 2 through 7 is

$$O((\mathsf{M}(n)\log n + n\log(n\log A))\mathsf{M}(n + \log A))$$

or $O^\sim(n^2 + n\log A)$ word operations.

15.29 Let $s_{ij}, t_{ij} \in R[x]$ be such that $s_{ij} f_i + t_{ij} f_j = 1$ for $1 \le i < j \le r$. Multiplying the congruences for $1 \le i \le k$ and $k < j \le r$, we obtain $s^*, t^* \in R[x]$ such that $s^* g + t^* h = 1$. Then we can take $s = s^* b^{-1}$ rem h and $t = t^* + (s^* b^{-1}$ quo $h)g$.

Chapter 16

16.2 (i) Let $F_k, F_k^* \in \mathbb{R}^{k \times n}$ consist of the first k rows of F and F^*, respectively, and M_k be the principal $k \times k$ submatrix of M. Then (2) implies that $F_k = M_k F_k^*$, we have $\det M_k = 1$, so that M_k is invertible, and this proves that the subspace $U_k \subseteq \mathbb{R}^n$ spanned by the rows of F_k is equal to the subspace spanned by the rows of F_k^*.

(ii) By (iii), f_k^* is orthogonal to $f_1^*, \ldots, f_{k-1}^*$, and since these span U_{k-1}, by (i), we conclude that $f_k^* \in U_{k-1}^\perp$. Now the claim follows from $f - f_k^* \in U_{k-1}$.

(iii) We prove by induction on k that $f_1^*, \ldots, f_k^*$ are pairwise orthogonal. The case $k = 1$ is trivial, and we assume that $k \ge 2$. For $1 \le l < k$, we have

$$f_k^* \star f_l^* = f_k \star f_l^* - \sum_{1 \le j < k} \mu_{kj} (f_j^* \star f_l^*) = f_k \star f_l^* - \frac{f_k \star f_l^*}{f_l^* \star f_l^*} f_l^* \star f_l^* = 0,$$

since $f_j^* \star f_l^* = 0$ for $j \ne l$, by the induction hypothesis.

(iv) is immediate from (2).

16.3 (ii) $f_0(x) = 1$, $f_1(x) = x$, $f_2(x) = x^2 - 1/4$, $f_3(x) = x^3 - x/2$.

16.9 Let $U = \mathbb{R}h_1 + \cdots + \mathbb{R}h_{i-2} = \mathbb{R}g_1 + \cdots + \mathbb{R}g_{i-2}$. In the proof of Lemma 16.13 (iii), we have seen that h_i^* is the component of g_{i-1}^* orthogonal to $U + \mathbb{R}g_i = U + \mathbb{R}h_{i-1}^*$. Now g_{i-1}^* is already orthogonal to U, and hence

$$h_i^* = g_{i-1}^* - \frac{g_{i-1}^* \star h_{i-1}^*}{h_{i-1}^* \star h_{i-1}^*} h_{i-1}^*.$$

The claim now follows by plugging in $h_{i-1}^* = g_i^* + \mu_{i,i-1} g_{i-1}^*$, which was shown in the proof of Lemma 16.13 (ii).

16.10 (i) We have

$$\begin{aligned} 0 \le \left\| \|y\|_2 x + \|x\|_2 y \right\|_2^2 &= (\|y\|_2 x + \|x\|_2 y) \star (\|y\|_2 x + \|x\|_2 y) \\ &= \|y\|_2^2 (x \star x) + 2\|x\|_2 \|y\|_2 (x \star y) + \|x\|_2^2 (y \star y) \\ &= 2\|x\|_2 \|y\|_2 (\|x\|_2 \|y\|_2 + x \star y). \end{aligned}$$

The claim is trivial if one of x and y is zero, and we may assume that $\|x\|_2 \|y\|_2 > 0$. Then $-x \star y \le \|x\|_2 \|y\|_2$. Replacing x by $-x$, we find that also

$$x \star y = -(-x \star y) \le \|-x\|_2 \|y\|_2 = \|x\|_2 \|y\|_2,$$

and the claim follows.

(ii) We calculate

$$\begin{aligned}\|x+y\|_2^2 &= (x+y)\star(x+y) = \|x\|_2^2 + 2(x\star y) + \|y\|_2^2 \\ &\le \|x\|_2^2 + 2\|x\|_2\|y\|_2 + \|y\|_2^2 = (\|x\|_2 + \|y\|_2)^2.\end{aligned}$$

In fact, it is also easy to deduce the Cauchy-Schwarz inequality from the triangle inequality.

16.12 In the 1999 edition, the text of the exercise contains several errors, and we first give a corrected version of it.

This exercises discusses basis reduction for polynomials. Let F be a field, $R = F[y]$, and $n \in \mathbb{N}_{>0}$. The **max-norm** of a vector $f = (f_1, \ldots, f_n) \in R^n$ is $\|f\| = \|f\|_\infty = \max\{\deg f_i : 1 \le i \le n\}$. For vectors $f_1, \ldots, f_m \in R$ which are linearly independent over $F(y)$, the field of fractions of R, the $\boldsymbol{R}$**-module** spanned by $f_1, \ldots, f_m$ is $M = \sum_{1\le i\le m} R f_i$, and $(f_1, \ldots, f_m)$ is a **basis** of M.

(i) Let $f_1, \ldots, f_m \in R^n$ be linearly independent (over $F(y)$), with $f_i = (f_{i1}, \ldots, f_{in})$ for $1 \le i \le m$. We say that the sequence $(f_1, \ldots, f_m)$ is **reduced** if

- $\|f_1\| \le \|f_2\| \le \cdots \le \|f_m\|$, and
- $\deg f_{ij} \le \deg f_{ii}$ for $1 \le j \le n$, with strict inequality if $j < i$, for $1 \le i \le m$.

In particular, we have $\|f_i\| = \deg f_{ii}$ for $1 \le i \le m$. Prove that f_1 is a **shortest vector** in the R-module $M = \sum_{1\le i\le m} R f_i$, so that $\|f_1\| \le \|f\|$ for all nonzero $f \in M$.

(ii) Consider the following algorithm, from von zur Gathen (1984a).

ALGORITHM 16.27 Basis reduction for polynomials.

Input: Linearly independent row vectors $f_1, \ldots, f_m \in R^n$, where $R = F[y]$ for a field F, with $\|f_i\| < d$ for $1 \le i \le m$.

Output: Row vectors $g_1, \ldots, g_m \in R^n$ and a permutation matrix $A \in R^{n\times n}$ such that $(g_1, \ldots, g_m)$ is a reduced sequence and $(g_1 A, \ldots, g_m A)$ is a basis of $M = \sum_{1\le i\le m} R f_i$.

1. let $g_1, \ldots, g_m$ be such that $\{g_1, \ldots, g_m\} = \{f_1, \ldots, f_m\}$ and $\|g_i\| \le \|g_{i+1}\|$ for $1 \le i < m$
   $A \longleftarrow \mathrm{id}$, $k \longleftarrow 1$
2. **while** $k \le m$ **do**
3. { $(g_1, \ldots, g_{k-1})$ is reduced and $\|g_i\| \le \|g_{i+1}\|$ for $1 \le i < m$ }
   $u \longleftarrow \|g_k\|$
4. **for** $i = 1, \ldots, k-1$ **do**
5. $q \longleftarrow g_{ki}$ quo g_{ii}, $g_k \longleftarrow g_k - q g_i$
6. **if** $\|g_k\| < u$ **then**
   $r \longleftarrow \min\{i : i = k \text{ or } (1 \le i < k \text{ and } \|g_i\| > \|g_k\|)\}$
   replace $g_r, \ldots, g_{k-1}, g_k$ by $g_k, g_r, \ldots, g_{k-1}$
   $k \longleftarrow r$, **goto** 2

7. $l \longleftarrow \min\{k \le j \le n : \deg g_{kl} = u\}$
   let $B \in R^{n\times n}$ be the permutation matrix for the exchange of columns k and l
   for $i = 1,\ldots,m$ **do** $g_i \longleftarrow g_i B$
   $A \longleftarrow BA, \quad k \longleftarrow k+1$
8. **return** $g_1,\ldots,g_m$ and A

Show that $M = \sum_{1\le i\le m} R\cdot g_i A$ holds throughout the algorithm, and conclude that the g_i are always nonzero vectors.

(iii) Assume that the invariants in curly braces are true in step 3. Convince yourself that $\|g_{k-1}\| \le u$ holds during steps 4 and 5 if $k \ge 2$. Show that $g_{ii} \ne 0$ holds in step 5 if $k \ge 2$, so that the division with remainder can be executed, and prove the invariants $\|g_k\| \le u$ and $\deg g_{kj} < u$ for $1 \le j < i$ of the loop 4.

(iv) Show that $(g_1,\ldots,g_{k-1})$ is reduced and $\|g_i\| \le \|g_{i+1}\|$ for $1 \le i < m$ holds each time the algorithm passes through step 3. Conclude that it works correctly if it halts in step 8.

(v) Show that $\|g_i\| < d$ for $1 \le i \le m$ holds throughout the algorithm. Prove that the cost for one execution of steps 3 through 7 is $O(nm)$ arithmetic operations (additions, multiplications, and divisions with remainder) in R or $O(nm\,\mathsf{M}(d))$ operations in F.

(vi) Show that the function $s(g_1,\ldots,g_m) = \sum_{1\le i\le m}\|g_i\|$ never increases in the algorithm and strictly decreases if the condition in step 6 is true. Conclude that the number of times when the latter happens is at most md and that the number of iterations of the loop 2 is at most $(m-1)(md+1)$.

(vii) Putting everything together, show that the running time of the algorithm is $O(nm^3 d\,\mathsf{M}(d))$ or $O^{\sim}(nm^3d^2)$ arithmetic operations in F.

(viii) Trace the algorithm on the $\mathbb{F}_{97}[y]$-module generated by

$$\begin{gathered}\left(5y^3+44y^2+37y+91,\ 8y^3+86y^2+91y+89,\ 16y^3+65y^2+20y+76\right),\\ \left(8y^3+70y+37,\ 16y^3+7y^2+54y+38,\ 32y^3+23y^2+80y+77\right),\\ \left(16y^2+84y+63,\ 32y^2+15y+19,\ 64y^2+48y+51\right) \in \mathbb{F}_{97}[y]^3.\end{gathered}$$

Solution:

(i) Let $f = \sum_{1\le i\le m} r_i f_i$ be a nonzero vector in M, with all $r_i \in F[y]$, let $e = \max\{\|r_i f_i\| : 1 \le i \le m\}$, and let l be the least index such that $\|r_l f_l\| = \deg(r_l f_{ll}) = e$. This implies in particular that $r_l \ne 0$. Then for $1 \le i < l$, we have

$$\deg(r_i f_{il}) \le \deg(r_i f_{ii}) = \|r_i f_i\| < \|r_l f_l\| = e,$$

and for $l < i \le m$, we find

$$\deg(r_i f_{il}) < \deg(r_i r_{ii}) = \|r_i f_i\| \le \|r_l f_l\| = e.$$

Thus the lth entry of f, which equals $\sum_{1\le i\le m} r_i f_{il}$, has degree e, and we conclude that

$$\|f_1\| \le \|f_l\| \le \deg r_l + \|f_l\| = \|r_l f_l\| = e \le \|f\|.$$

We note that reduced bases are not unique: for example, both $(1,0),(0,1)$ and $(1,1),(0,1)$ are reduced bases of the R-module R^2.

(ii) The invariant clearly holds initially, so let us assume that it holds at some pass of the algorithm through step 2. There are three points in the algorithm where the g_i's change. In step 5, a polynomial multiple of g_i is added to g_k. This is an invertible transformation and does not change the generated module. In step 6, the g_i's are permuted, which does not change the spanned module either. In step 7, all g_i's are multiplied by the permutation matrix B, and the invariant holds again at the next pass through step 2 since A is multiplied by the inverse of B, which happens to be B itself. Thus the invariant holds at all times in the algorithm, the g_i's are always linearly independent, and nonzero in particular.

(iii) Since $(g_1,\ldots,g_{k-1})$ is reduced, we have $\|g_i\| = \deg g_{ii}$ and g_{ii} is nonzero, by (ii). Before the first iteration of the loop 4, we have $\|g_{k-1}\| \le \|g_k\| = u$. Since g_{k-1} does not change in steps 4 and 5, $\|g_{k-1}\| \le u$ holds throughout these steps. So we assume that $\|g_k\| \le u$ holds before some pass through step 5. There is nothing to prove if $q = 0$, and otherwise, we find

$$\begin{aligned} \deg q &= \deg g_{ki} - \deg g_{ii} \le \|g_k\| - \|g_i\|, \\ \|g_k - q g_i\| &\le \max\{\|g_k\|, \deg q + \|g_i\|\} \le \max\{\|g_k\|, \|g_k\|\} = u, \end{aligned}$$

so that $\|g_k\| \le u$ holds again after step 5.

The second invariant is vacuously true at the beginning of the loop, and we assume that it holds before some pass through step 5. If $j < i$, then $\deg g_{ij} < \deg g_{ii} = \|g_i\| \le \|g_{k-1}\|$, since $(g_1,\ldots,g_{k-1})$ is reduced, and

$$\begin{aligned} \deg(g_{kj} - q g_{ij}) &\le \max\{\deg g_{kj}, \deg q + \deg g_{ij}\} < \max\{u, \deg q + \deg g_{ii}\} \\ &= \max\{u, \deg g_{ki}\} = u, \end{aligned}$$

by the first invariant. This inequality also holds if $q = 0$, and together with

$$\deg(g_{ki} - q g_{ii}) = \deg(g_{ki} \text{ rem } g_{ii}) < \deg g_{ii} = \|g_i\| \le \|g_{k-1}\| \le u$$

implies that the invariant holds again before the next pass through step 5.

(iv) The invariants are clearly true before the first pass through step 2, and we assume that they hold before step 3. The $g_1,\ldots,g_{k-1}$ do not change in steps 3 through 5, so that the first invariant holds again at the next pass through step 3 if the condition in step 6 is true, and otherwise this is ensured by (iii) and the actions taken in step 7. Moreover, the g_i's are resorted in step 6 when the if condition is true, and hence the second invariant holds again after step 6 and at the next pass through step 3. In particular, $(g_1,\ldots,g_m)$ is reduced if the algorithm terminates in step 8.

(v) The value $\max\{\|g_i\| : 1 \le i \le m\}$ is less than d initially, it changes only during steps 4 and 5, and (iii) implies that one complete pass through these steps does not increase the value. Thus the degree of all polynomials in the algorithm is less than d, at all times. The cost for one execution of step 5 is $O(n)$ arithmetic operations in R, each taking $O(\mathsf{M}(d))$ operations in F, and there are $O(m)$ iterations of the loop 4. All other steps are for free, and hence the cost for one iteration of the loop 2 is $O(nm\,\mathsf{M}(d))$ field operations.

(vi) By a similar argument as in (v), s changes only during steps 4 and 5, and a complete pass through these steps does not increase its value. It decreases strictly between two successive passes through step 3 if and only if the condition in step 6 is true. Since all g_i's are nonzero, by (ii), s is always a nonnegative integer, and initially $s < md$. Thus the number of decreases of s is at most md. Between each two times that the condition in step 6 is true, there are at most $m-1$ passes of the loop 2 where the condition is false, and also before the first and after the last decrease. Thus the total number of iterations of the loop 2 is at most $(m-1)\cdot(md+1)$.

16.13 The analog statement is as follows. Let F be a field and $R = F[y]$, let $f, g \in F[x,y] = R[x]$ have positive degrees n, k in x, respectively, and suppose that $u \in R[x]$ is monic nonconstant with respect to x and divides both f and g modulo m for some $m \in R$ with $k\deg_y f + n\deg_y g < \deg_y m$. Then $\gcd(f,g) \in R[x]$ is nonconstant with respect to x.

We imitate the proof of Lemma 16.20, and suppose that $\gcd(f,g) = 1$ in $F(y)[x]$. Then there exist $s,t \in R[x]$ such that $sf + tg = \operatorname{res}_x(f,g)$, by Corollary 6.21. Since u divides both f and g modulo m, it divides $\operatorname{res}_x(f,g) \in R$ modulo m. With respect to x, the polynomial u is monic and nonconstant, and thus $\operatorname{res}_x(f,g) \equiv 0 \bmod m$. Since $\deg_y(\operatorname{res}_x(f,g)) \le k\deg_y f + n\deg_y g < \deg_y m$, by Theorem 6.22, it follows that $\operatorname{res}_x(f,g)$ is zero. This contradiction to our assumption shows that $\gcd(f,g) \in F(y)[x]$ is nonconstant. By Corollary 6.10, the gcd of f and g in $R[x]$ is also nonconstant.

16.15 The number of nonzero coefficients of the polynomial is 2^n. The arithmetic circuit first computes y^{2^i} for $1 \le i < n$, taking $n-1$ squarings, then computes $x + y^{2^i}$ for $0 \le i < n$, taking n additions, and finally multiplies all factors up, taking another $n-1$ multiplications.

16.16 (i) It is clear that σ is a ring homomorphism, so that in particular its restriction to U is F-linear. Moreover, we have $\deg(\sigma(f)) \le (n-1)\sum_{1\le i\le t} n^{i-1} = n^t - 1$ for $f \in U$, so that $\sigma(U) \subseteq V$. Moreover, the monomials $B = \{x_1^{e_1}\cdots x_t^{e_t} : 0 \le e_1,\ldots,e_t < n\}$ form an F-basis of U, and $\sigma(B) = \{x^i : 0 \le i < n\}$ is an F-basis of V, so that σ maps U isomorphically onto V.

(ii) Let $R = F[x_1,\ldots,x_t]$ for short. If $g \in R$ is a factor of f, then, by unique factorization in $F[x]$, there is a unique subset $S \subseteq \{1,\ldots,r\}$ and a nonzero constant

$c \in F$ such that $\sigma(g) = c\prod_{i\in S} h_i = ch$. Since g divides f, it has degree less than n in all variables, and hence $h \in V$ and $\sigma^{-1}(h) = c^{-1}g$ divides f.

(iii) Let $h_1,\ldots,h_r \in F[x]$ be as in (ii), and $S \subseteq \{1,\ldots,r\}$. Then $r < n^t$, and we can compute $h = \prod_{i\in S} h_i$ and $h^* = \sigma(f)/h$ in time $O(\mathsf{M}(n^t)t\log n)$. Computing $g = \sigma^{-1}(h)$ and $g^* = \sigma^{-1}(h^*)$ is for free. Since $\sigma(x_1) = x$, we may identify x_1 and x. Then $F[x]$ becomes a subring of R, and $f \equiv gg^* \bmod I$, where $I = \langle x_t - x^{n^{t-1}},\ldots,x_2 - x^n\rangle \subseteq R$. Finally, we check whether $f = gg^*$ holds, at a cost of $O(\mathsf{M}(n)^t)$ operations. This solves our task completely.

There is, however, a way to avoid the computation of $g\cdot g^*$ if we proceed as follows to compute g and g^*. We define $f_i \in F[x_1,x_{i+1},\ldots,x_t]$ by $f_i = f(x,x^n,\ldots,x^{n^{i-1}}, x_{i+1},\ldots,x_t)$, and g_i,g_i^* are derived from g,g^* in a similar way, for $1 \le i \le t$. Then $f_1 = f$, $g_1 = g$, $g_1^* = g^*$, $f_t = \sigma(f)$, $g_t = h$, and $g_t^* = h^*$. The degrees in x of f_i,g_i,g_i^* are less than n^i, for all i, and all these polynomials can be computed without arithmetic cost. We now claim that $f = gg^*$ if and only if $\deg_x f_i = \deg_x(g_ig_i^*)$ for all i. The "only if" part is clear. For the converse, we show by reverse induction on i that $f_i = g_ig_i^*$. This is clear for $i = t$, and we assume that $i < t$. The induction hypothesis implies that

$$f_i \equiv f_{i+1} = g_{i+1}g_{i+1}^* \equiv g_ig_i^* \bmod x^{n^i} - x_{i+1}.$$

By assumption, both sides have degree less than n^i in x, and hence they are equal.

Thus it is sufficient to check whether all g_i,g_i^* have correct degrees in x, and no multivariate multiplications are necessary. The number of subsets S that have to be checked in the worst case is 2^{n^t}, and hence the overall cost, without the cost for the univariate factorization, to find one irreducible factor of f is $O(t2^{n^t}\mathsf{M}(n^t)\log n)$ field operations. This is singly exponential in the degree and doubly exponential in the number of variables.

If one wanted to use this in practice, one would first check probabilistically if $f = gg^*$ holds, say by substituting $x_1 - a_i$ for x_i, with random $a_2,\ldots,a_n$.

(In the 1999 edition, some variables were named differently.)

Chapter 17

17.1 (i) The encryption of "ALGEBRAISFUN" is

$$8100,\ 8019,\ 14487,\ 96,\ 15989,\ 10786.$$

(ii) We have

$$t \equiv \sum_{0\le i\le 9} w^{-1}x_ia_i \equiv \sum_{0\le i\le 9} x_ic_i \bmod m,$$

and the claim follows since both sides of the congruence are nonnegative and less than m.

(iii) The original message was "LATTICEREDUCTION".

Chapter 18

18.1 If a is even, then $a^2 \equiv 0 \bmod 4$, and if a is odd, then $a^2 \equiv 1 \bmod 4$.

18.2 We have $\varphi(55) = \varphi(5) \cdot \varphi(11) = 40$, and Euler's theorem implies that $2^{40} \equiv 1 \bmod 55$. Thus

$$2^{1\,000\,005} = (2^{40})^{25\,000} \cdot 2^5 \equiv 2^5 = 32 \bmod 55.$$

18.3 $N = 10^{200} + 349$ is composite since $2^{N-1} \not\equiv 1 \bmod N$.

18.6 (i) Let $N = p^e$ for an odd prime p and $e \geq 1$. Then Fermat's little theorem says that $a^{p-1} \equiv 1 \bmod p$, and since $(p-1) \mid (p^e - 1) = N - 1$, we also have $a^{N-1} \equiv 1 \bmod p$. Thus $\gcd(a^{N-1} - 1, N)$ is divisible by p.

(ii) The criterion is as follows. Let $N - 1 = 2^v m$ with $v, m \in \mathbb{N}$ and m odd. If either $\gcd(a^{N-1} - 1, N) = 1$ or there is an index $i \in \{1, \ldots, v\}$ such that $a^{2^i m} \equiv 1 \bmod N$ and $a^{2^{i-1} m} \not\equiv \pm 1 \bmod N$, then N is not a prime power. To see why, we let N be as in (i). Then (i) implies that $\gcd(a^{N-1} - 1, N) > 1$. If $a^{2^i m} \equiv 1 \bmod N$ for some $i \leq v$, then $a^{2^{i-1} m}$ is a square root of 1 modulo N. By Exercise 9.40, the only square roots of 1 modulo an odd prime power are 1 and -1, and hence $a^{2^{i-1} m} \equiv \pm 1 \bmod p$.

In fact, if N is composite and $a^{2^{i-1} m} \not\equiv \pm 1 \bmod N$, then $\gcd(a^{2^{i-1} m} - 1, N)$ is a nontrivial factor of N.

18.9 (i) By Lemma 18.4, we may assume that N is squarefree. Let p be a prime divisor of N and $b \in \mathbb{Z}$ coprime to p with $\operatorname{ord}_p(b) = p - 1$; such a b exists according to Exercise 8.16. By the Chinese Remainder Theorem 5.3, there exists an $a \in \mathbb{Z}$ such that $a \equiv b \bmod p$ and $a \equiv 1 \bmod N/p$. Then $\gcd(a, N) = 1$ and $a^{N-1} \equiv 1 \bmod p$, and Lemma 18.1 implies that $p - 1$ divides $N - 1$.

(ii) "$\Longrightarrow$" follows from Lemma 18.4 and (i).

"$\Longleftarrow$": Let N be squarefree and enjoy the property in (i), and $a \in \mathbb{Z}_N^\times$. Then $a^{p-1} \equiv 1 \bmod p$, and hence $a^{N-1} \equiv 1 \bmod p$, for all prime divisors p of N. Thus $a^{N-1} \equiv 1 \bmod N$.

(iii) If the Carmichael number N were even, then, since it is squarefree by Lemma 18.4 and composite, it would have an odd prime divisor p, and the even number $p - 1$ would divide the odd number $N - 1$. Now we assume that $p < q$ are odd primes with $N = pq$. Then $q - 1$ divides $N - 1 = pq - 1$, by (i), and hence it also divides $(pq - 1) - p(q - 1) = p - 1 < q - 1$. This contradiction proves the claim.

(iv) $561 = 3 \cdot 11 \cdot 17$, $1105 = 5 \cdot 13 \cdot 17$, $1729 = 7 \cdot 13 \cdot 19$, $2465 = 5 \cdot 17 \cdot 29$, $2821 = 7 \cdot 13 \cdot 31$, $172\,081 = 7 \cdot 13 \cdot 31 \cdot 61$ are the only Carmichael numbers in the list. $663 = 3 \cdot 13 \cdot 17$ has $13 - 1 \nmid 663 - 1$, $867 = 3 \cdot 17^2$ is not squarefree, $935 = 5 \cdot 11 \cdot 17$ has $11 - 1 \nmid 935 - 1$, 1482 is even, $1547 = 7 \cdot 13 \cdot 17$ has $7 - 1 \nmid 1547 - 1$, $2077 = 31 \cdot 67$ has only two prime factors, and 2647 is prime.

18.11 (i) The proper divisors of P_n are $1,2,4,\ldots,2^{n-1}$ and $M_n, 2M_n, 4M_n, \ldots, 2^{n-2}M_n$. The first ones sum to $2^n - 1 = M_n$, and the last ones to $(2^{n-1}-1)M_n$, and hence the sum of all proper divisors is $2^{n-1}M_n = P_n$.

(iii) Multiplying out the product defining N, we find

$$\begin{aligned} N &= \prod_{1\le i\le 2n-1} (d_i P_n m + 1) = 1 + P_n m \sum_{1\le i\le 2n-1} d_i + l P_n^2 m^2 \\ &= 1 + P_n^2 m(1 + lm) \end{aligned}$$

for some $l \in \mathbb{N}$. Thus $p_i - 1 = d_i P_n m \mid P_n^2 m \mid N - 1$ for all i.

18.12 (ii) ALGORITHM 18.16 Special integer factorization.

Input: A squarefree odd integer $N \ge 3$, a multiple $L \in \mathbb{N}$ of $\lambda(N)$, and a confidence parameter $k \in \mathbb{N}$.

Output: The set of prime divisors of N.

1. $h \longleftarrow 1$
2. **while** $h < k$ **do**
3. choose $a \in \{2,\ldots,N-2\}$ uniformly at random
4. $g \longleftarrow \gcd(a,N)$
   if $g > 1$ **then** break the loop 2 and **goto** 9
5. write $L = 2^v m$ with $v, m \in \mathbb{N}$, $v \ge 1$, and m odd
   call the repeated squaring algorithm 4.8 to compute $b_0 = a^m \bmod N$
   if $b_0 = 1$ **then** $h \longleftarrow h+1$, **goto** 2
6. **for** $i = 1,\ldots,v$ **do** $b_i \longleftarrow b_{i-1}^2 \bmod N$
7. $j \longleftarrow \max\{0 \le i \le v : b_i \ne 1\}$, $g \longleftarrow \gcd(b_j + 1, N)$
   if $g = 1$ or $g = N$ **then** $h \longleftarrow h+1$
   else break the loop 2 and **goto** 9
8. **return** $\{N\}$
9. **call** the algorithm recursively with input g, L, k and with input $N/g, L, k$ to compute the sets U, V of prime factors of g and N/g, respectively
   return $U \cup V$

(The numbers L and m were called m and m^*, respectively, in the 1999 edition.)

It is clear that the algorithm returns a (possibly incomplete) factorization of N if it terminates. We claim that if N is composite, then $1 < g < N$ holds in step 4 or 7 with probability at least $1/2$ over the random choices in step 2. This implies that each returned factor is prime with probability at least $1 - 2^{-k}$, and that the returned factorization is the prime factorization with probability at least $1 - r2^{-k}$, where $r \le \log_2 N$ is the number of returned factors.

To prove the claim, we proceed as in the proof of Theorem 18.6, and let P be the set of prime divisors of N and

$$I = \{i : 0 \le i \le v \text{ and } \forall u \in \mathbb{Z}_N^\times \; u^{2^i m} = 1\}.$$

Since $\lambda(N) \mid L$, we have $v \in I$. As in the proof of Theorem 18.6, we find that $0 \notin I$, let $l < v$ be such that $l \notin I$ and $l+1 \in I$, and

$$G = \{u \in \mathbb{Z}_N^\times : u^{2^l m} = \pm 1\} \subseteq \mathbb{Z}_N^\times .$$

Then $\#G < \varphi(N)/2$ since N is composite. Now assume that $a \in \mathbb{Z}_N^\times \setminus G$ in step 3. As in the proof of Theorem 18.6, we find that $j = l$ in step 7 and

$$g = \gcd(b_l + 1, N) = \prod_{\substack{p \in P \\ a^{2^l m} \equiv -1 \bmod p}} p$$

is a proper divisor of N. Since $\#(\mathbb{Z}_N^\times \setminus G) \geq (\#\mathbb{Z}_N)/2$ and the algorithm detects a proper factor of N in step 4 if $a \in \mathbb{Z}_N \setminus \mathbb{Z}_N^\times$, we obtain the claimed probability bound.

The cost for one iteration of the loop 2 is $O((\log L + \operatorname{loglog} N)\mathsf{M}(\log N))$ word operations, or $O(\log N \cdot \mathsf{M}(\log N))$ if $\log L \in O(\log N)$. By what we have just shown, the expected number of iterations until a proper factor is found is at most 2 if N is composite, and k if N is prime. Since the sum of the word lengths over all leaves of the recursion tree is $O(\log N)$, the total cost for all leaves is $O(k \log N \cdot \mathsf{M}(\log N))$ word operations, by the superlinearity of M. The sum of the word lengths of the inner vertices at each level of the tree is $O(\log N)$ as well, the depth of the recursion tree is at most $r \in O(\log N)$, and hence the expected total cost for all inner vertices of the tree is $O(\log^2 N \cdot \mathsf{M}(\log N))$ word operations. Thus the worst case overall cost is $O(k \log^2 N \cdot \mathsf{M}(\log N))$ word operations, while the expected cost is only $O((k + \log N) \log N \cdot \mathsf{M}(\log N))$.

18.13 (i) For $1 \leq i \leq r$, we have $b^{\varphi(p_i^{e_i})} \equiv 1 \bmod p_i^{e_i}$, by Euler's theorem, for all $b \in \mathbb{Z}$ not divisible by p_i. Since $\varphi(p_i^{e_i}) \mid \lambda(N)$, we conclude that $b^{\lambda(N)} \equiv 1 \bmod p_i^{e_i}$ if $p_i \nmid b$, and hence $b^{\lambda(N)} \equiv 1 \bmod N$ if $\gcd(b, N) = 1$.

(ii) We conclude from (i) that $a^{N-1} = 1$ for all $a \in \mathbb{Z}_N^\times$ if $\lambda(N) \mid N-1$. For the converse, we assume that $a^{N-1} = 1$ for all $a \in \mathbb{Z}_N^\times$. If N is prime, then $\lambda(N) = N-1$. Otherwise, N is a Carmichael number, and Exercise 18.9 (ii) shows that N is squarefree and $p-1 \mid N-1$ for all prime divisors p of N. But $\lambda(N)$ is the least common multiple of all these $p-1$, and therefore $\lambda(N) \mid N-1$.

18.15 (i) If $b \in \mathbb{F}_p^\times$ is a square, then Lemma 14.7 says that $b^{(p-1)/2} = 1$. Thus $\operatorname{ord}(b) \leq (p-1)/2 < p-1$, and hence b does not generate $\mathbb{F}_p^\times$.

(ii) Let $b \in \mathbb{F}_p^\times$ be a nonsquare. Then Lemma 14.7 implies that $b^{(p-1)/2} = -1$, and since 2 is the only prime divisor of $p-1$, Exercise 8.16 (i) shows that $\operatorname{ord}(b) = p-1$ and b generates $\mathbb{F}_p^\times$.

18.16 (i) Let $b = a^r$. Then $b^{2^s} \equiv a^{p-1} \equiv 1 \bmod p$, by Fermat's little theorem, and $b \bmod p$ is a 2^sth root of unity. Since p is prime and $2^s < p$, 2^s is a unit modulo p. Finally, since $a \bmod p$ is a nonsquare, Lemma 14.7 shows that $b^{2^{s-1}} \equiv a^{(p-1)/2} \equiv -1 \bmod p$, and hence $b \bmod p$ is a primitive 2^sth root of unity.

(ii) The algorithm chooses $a \in \{2,\ldots,p-1\}$ uniformly at random and checks whether $a^{(p-1)/2} \equiv -1 \bmod p$ by repeated squaring. If this is not the case, then it chooses another a. Otherwise, it returns a^r rem p. The cost for the repeated squaring, which can be arranged such that a^r rem p is computed along the way, is $O(\log p)$ multiplications modulo p or $O(\log p \cdot \mathsf{M}(\log p))$ word operations. Since at least half of the elements between 2 and $p-1$ are nonsquares modulo p, by Lemma 14.7, the expected number of iterations of the algorithm is at most 2.

(iii) The integer $2^{27}k+1$ is prime for $k \in \{15,17,24,26,29\}$ and composite for all other values of k between 1 and 31. Thus there are precisely four primes $2^{31} < p < 2^{32}$ such that $2^{27} \mid p-1$ and only two with $2^{28} \mid p-1$.

18.18 (i) Let $x \geq 59$. From the prime number theorem 18.7, we have

$$\begin{aligned}\pi(2x)-\pi(x) &\geq \frac{2x}{\ln 2x}\left(1+\frac{1}{2\ln 2x}\right)-\frac{x}{\ln x}\left(1+\frac{3}{2\ln x}\right) \qquad (4)\\ &\geq \frac{x}{\ln x}\left(\frac{2\ln x}{\ln x+\ln 2}-1-\frac{3}{2\ln x}\right)=\frac{x}{\ln x}\left(\frac{2}{1+\frac{\ln 2}{\ln x}}-1-\frac{3}{2\ln x}\right)\\ &\geq \frac{x}{\ln x}\left(2\left(1-\frac{\ln 2}{\ln x}\right)-1-\frac{3}{2\ln x}\right) \geq \frac{x}{\ln x}\left(1-\frac{3}{\ln x}\right)\\ &\geq \frac{x}{2\ln x},\end{aligned}$$

where we have used that $(1+\delta)^{-1} \geq 1-\delta$ for $0<\delta<1$ and $2\ln 2+3/2 \approx 2.89 < 3$ in the third line, and the last inequality holds if and only if $x \geq e^6 \approx 403.43$.

(iii) Using (i), we have $\pi(2^{32})-\pi(2^{31}) \geq 2^{31}/(61\ln 2) > 49\,970\,387$ and $\pi(2^{64})-\pi(2^{63}) \geq 2^{63}/(126\ln 2) > 1.056\cdot 10^{17}$. In fact, the more accurate estimate (4) implies that there are more than 91 082 775 32-bit primes and more than $2.02\cdot 10^{17}$ 64-bit primes.

(iv) We need r single precision primes such that their product exceeds $2C = 2n^{n/2}B^n \approx 2n^{n/2}2^{n^2} = 2^{n^2+(n\log n)/2+1}$, where log denotes the binary logarithm. Each single precision prime is greater than 2^{k-1}, so that $(k-1)r \geq n^2+(n\log n)/2+1$ is sufficient. For $k=32$ and $k=64$, the number of single precision primes is at least $9\cdot 10^7$ and $2\cdot 10^{17}$, respectively, and substituting these numbers for r leads—after some calculation—to admissible values of n up to (at least) 52 816 and 3 549 647 861, respectively.

18.19 (i) In Table 18.2, the first column contains the value of s, the second column the estimated number $\lfloor 2^{k-s}/(k-1)\ln 2 \rfloor$ of Fourier primes, and the third column the true number of Fourier primes for that value of s.

(ii) To multiply two polynomials of degree less than $n = 2^{s-1}$ with coefficients of bit length at most $l = 2^{s-1}$, we need r Fourier primes such that their product exceeds $n2^{2l+1} = 2^{s+2^s}$. Since each prime is greater than 2^{k-1}, $r \geq (s+2^s)/(k-1)$ primes are sufficient. Now the estimate for the number of Fourier primes implies

$k=32$			$k=64$		
1	99940774	98182656	33	49177206	48742226
2	49970387	49090415	34	24588603	24371651
3	24985193	24545135	35	12294301	12184774
4	12492596	12273201	36	6147150	6092470
5	6246298	6136376	37	3073575	3044704
6	3123149	3068306	38	1536787	1522110
7	1561574	1534382	39	768393	761041
8	780787	766507	40	384196	380158
9	390393	382950	41	192098	189935
10	195196	191549	42	96049	94895
11	97598	95658	43	48024	47179
12	48799	47700	44	24012	23606
13	24399	23893	45	12006	11888
14	12199	12052	46	6003	6003
15	6099	6046	47	3001	2986
16	3049	3020	48	1500	1498
17	1524	1540	49	750	743
18	762	762	50	375	380
19	381	394	51	187	196
20	190	199	52	93	88
21	95	102	53	46	49
22	47	56	54	23	22
23	23	24	55	11	14
24	11	14	56	5	8
25	5	8	57	2	5
26	2	6	58	1	1
27	1	4	59	0	1
28	0	2	60	0	0
29	0	1	61	0	0
30	0	1	62	0	0
31	0	0	63	0	0

TABLE 18.2: Estimated and true number of Fourier primes (Exercise 18.19)

that $r \le 2^{k-s}/(k-1)\ln 2$, and combining both inequalities, we obtain the constraint $2^s(2^s+s) \le 2^k/\ln 2$. Substituting $k=32$ and $k=64$, we find that the maximal possible values for s are 16 and 32, respectively.

18.20 (This solution refers to the 2003 edition only.)

(i) One execution of Algorithm 19.2 takes $O(\mathsf{M}(r^{1/2})\mathsf{M}(\beta)(\log r+\log\beta))$ word operations, by Theorem 19.3, and we expect to make $O(\beta)$ choices before termination.

(ii) This follows from the fact that a number p which passes the gcd test is prime with probability at least $(B/2\beta)/(B/\ln r) = (\ln r)/2\beta$, and a similar analysis as in the proof of Theorem 18.8.

(iii) The expected number of primality tests is $O(k\beta/\ln r)$, taking $O(\beta\mathsf{M}(\beta))$

word operations each, and the claim follows from

$$\mathsf{M}(r^{1/2})(\log r + \log \beta) = 2\mathsf{M}(\beta^{1/2}) \log \beta \in O(\beta/\log \beta)$$

by adding up the costs.

18.21 We have $\vartheta(x) > x(1-1/\ln x) > x/2$ if $x \geq 49$, and we easily check that in fact $\vartheta(x) > x/2$ for $x \geq 5$.

(i) In step 1 of the small primes modular determinant algorithm 5.10, we let $x = \lceil 2\ln(2C+1) \rceil$ and choose the first $r = \pi(x)$ primes $2 = m_0 < m_1 < \cdots m_{r-1} \leq x$. Then $\prod_{0 \leq i < r} m_i = e^{\vartheta(x)} > 2C$, and the determinant is correctly recovered in step 4. Step 1 takes $O(r \log^2 r \log\log r)$ word operations, by Theorem 18.10 (ii). The cost for step 2 is $O(n^2 \log m_i \cdot \log B)$ for each i, in total $O(n^2 x \log B)$. The cost for step 3 is $O(n^3 \log^2 m_i)$ word operations for each i, in total $O(n^3 x \log x)$. Finally, the Chinese remaindering takes $O(x^2)$ word operations. Using $x \in O(n \log(nB))$ and $r \in O(x/\log x)$, we find an overall cost of $O(n^4 \log(nB) \log(n \log B) + n^3 \log^2(nB))$ word operations.

In step 1 of the small primes modular EEA 6.57, we let $x = \lceil 2\ln(2A^2B^3+1) \rceil$ and choose as S the first $r = \pi(x)$ primes. Similarly as above, we have $\prod_{p \in S} p > 2A^2B^3$ in step 1, $\prod_{p \in S} p > 2B^3$ in step 2, and $\prod_{p \in S_i} p > 2B^2$ in step 3, for each i, and the correctness follows as in the proof of Theorem 6.58. The cost for choosing the primes is $O(r \log^2 r \log\log r)$ word operations. Reducing all coefficients of f and g modulo all primes in S in step 2 takes $O(nx \log B)$ word operations, and the cost for all modular EEAs is $O(nmx \log x)$. Finally, the rational number reconstruction in step 3 takes $O(x^2)$ word operations per coefficient, together $O(nmx^2)$ word operations. Using $x \in O(n \log(nA))$ and $r \in O(x/\log x)$, we see that the cost for step 3 is dominant and obtain a total cost of $O(n^3 m \log^2(nA))$ word operations.

(ii) Let $\sigma = \operatorname{res}(f, f')$ and $C = (n+1)^{2n} A^{2n-1}$, such that $0 < |\sigma| < C$ and $\ln C \leq \gamma$. If we let $x = \lceil 2\ln C \rceil$ and $r = 2\pi(x)$, then the product of the first $\pi(x)$ primes is $e^{\vartheta(x)} \geq C > |\sigma|$, and hence any product of $\pi(x)$ of the first r primes exceeds $|\sigma|$. Thus at most $\pi(x) = r/2$ of the first r primes divide σ. By Theorem 18.10 (ii), the cost for computing the first r primes is $O(r \log^2 r \log\log r)$ word operations, or $O(\gamma \log \gamma \log\log \gamma)$ since $r \in O(\gamma/\log \gamma)$.

18.22 (i) Let $a \in \mathbb{F}_p^\times$ such that $a^2 = -1$. Then $a^4 = 1$ and $\operatorname{ord}(a) = 4$, by Exercise 8.16 (i). Thus $4 \mid \#\mathbb{F}_p^\times = p-1$, by Lagrange's theorem.

(ii) If $a \in \mathbb{F}_p^\times$ is not a square, then $a^{(p-1)/2} = -1$, by Lemma 14.7. Thus $a^{(p-1)/4}$ is a square root of -1.

18.23 (i) Let $p_1, \ldots, p_r$ be the distinct prime divisors of M and N, and $e_1, f_1, \ldots, e_r, f_r \in \mathbb{N}$ such that $N = \prod_{1 \leq i \leq r} p_i^{e_i}$ and $M = \prod_{1 \leq i \leq r} p_i^{f_i}$. Then

$$\left(\frac{ab}{N}\right) = \prod_{1 \leq i \leq r} \left(\frac{ab}{p_i}\right)^{e_i} = \prod_{1 \leq i \leq r} \left(\frac{a}{p_i}\right)^{e_i} \left(\frac{b}{p_i}\right)^{e_i} = \left(\frac{a}{N}\right)\left(\frac{b}{N}\right),$$

$$\left(\frac{a}{MN}\right) = \prod_{1 \leq i \leq r} \left(\frac{a}{p_i}\right)^{e_i+f_i} = \prod_{1 \leq i \leq r} \left(\frac{a}{p_i}\right)^{e_i} \left(\frac{a}{p_i}\right)^{f_i} = \left(\frac{a}{M}\right)\left(\frac{a}{N}\right).$$

(ii) Let $N = p_1 \cdots p_r$ and $a = q_1 \cdots q_s$, where $p_1, \ldots, p_r, q_1, \ldots, q_s \in \mathbb{N}$ are (not necessarily distinct) odd primes and $\{p_1, \ldots, p_r\} \cap \{q_1, \ldots, q_s\} = \emptyset$. Let $1 \leq i \leq r$ and $1 \leq j \leq s$. The law of quadratic reciprocity for the Legendre symbol says that $(\frac{p_i}{q_j}) = (\frac{q_j}{p_i})$ unless and only unless $p_i \equiv q_j \equiv 3 \bmod 4$. If we let $\varepsilon(u,v) = (-1)^{(u-1)(v-1)/4}$ for $u, v \in \mathbb{Z}$, then this can be rewritten as

$$\Big(\frac{p_i}{q_j}\Big) = \varepsilon(p_i, q_j)\Big(\frac{q_j}{p_i}\Big).$$

Let u, v, w be odd integers. Then $vw - 1 \equiv v + w - 2 \bmod 4$ and

$$\varepsilon(u, vw) = (-1)^{(u-1)(vw-1)/4} = \Big((-1)^{(u-1)/2}\Big)^{(v-1)/2+(w-1)/2} = \varepsilon(u,v)\varepsilon(u,w).$$

Thus, by symmetry, ε is multiplicative with respect to both (odd) arguments, and hence

$$\begin{aligned}\Big(\frac{N}{a}\Big) &= \prod_{\substack{1\leq i\leq r\\ 1\leq j\leq s}} \Big(\frac{p_i}{q_j}\Big) = \prod_{\substack{1\leq i\leq r\\ 1\leq j\leq s}} \varepsilon(p_i, q_j)\Big(\frac{q_j}{p_i}\Big) \\ &= \varepsilon(p_1 \cdots p_r, q_1 \cdots q_s) \prod_{\substack{1\leq i\leq r\\ 1\leq j\leq s}} \Big(\frac{q_j}{p_i}\Big) = \varepsilon(N, a)\Big(\frac{a}{N}\Big),\end{aligned}$$

and the claim follows.

(iii) Let $N = p_1 \cdots p_r$ as in (ii). Then $(\frac{2}{p_i}) = (-1)^{(p_i^2-1)/8} = \delta(p_i)$. Since the congruence $(vw)^2 - 1 \equiv v^2 - 1 + w^2 - 1 \bmod 16$ holds for all odd integers v, w, we find that δ is multiplicative on odd arguments and

$$\Big(\frac{2}{N}\Big) = \prod_{1\leq i\leq r} \Big(\frac{2}{p_i}\Big) = \prod_{1\leq i\leq r} \delta(p_i) = \delta(N).$$

(iv) Let $N = p_1 \cdots p_r$ as in (ii) and $b = a \text{ rem } N$. Then $a \equiv b \bmod p_i$ implies that $(\frac{a}{p_i}) = (\frac{b}{p_i})$ for all i, and hence

$$\Big(\frac{a}{N}\Big) = \prod_{1\leq i\leq r} \Big(\frac{a}{p_i}\Big) = \prod_{1\leq i\leq r} \Big(\frac{b}{p_i}\Big) = \Big(\frac{b}{N}\Big).$$

(v) ALGORITHM 18.17 Jacobi symbol computation.
Input: An odd integer $N > 1$ and $a \in \{1, \ldots, N-1\}$.
Output: The Jacobi symbol $\Big(\frac{a}{N}\Big)$.

1. write $a = 2^k b$ for $k, b \in \mathbb{N}$ and b odd
2. **if** $b = 1$ **then return** $(-1)^{k(N^2-1)/8}$

3. $M \longleftarrow N \bmod b$, **if** $M = 0$ **then return** 0
4. **call** the algorithm recursively to compute $u = \left(\frac{M}{b}\right)$
5. **return** $(-1)^{k(N^2-1)/8+(N-1)(b-1)/4}u$

The algorithm returns the correct result in step 2 if and only if a is a power of two, by (i) and (iii). If the algorithm returns 0 in step 3, then $\gcd(a,N) > 1$, and the output is correct as well. Otherwise, since b is odd, $1 < b < N$, and $0 < M < b$, we may conclude by induction that the result of the recursive call in step 4 is correct. If $\gcd(M,b) > 1$, then $\gcd(a,N) > 1$, $u = 0$, and the algorithm correctly returns 0 in step 5. Now assume that $\gcd(M,b) = 1$. Then $\gcd(a,N) = 1$, and using parts (i) through (iv) of this exercise, we find that

$$\begin{aligned}\left(\frac{a}{N}\right) &= \left(\frac{2}{N}\right)^k \left(\frac{b}{N}\right) = (-1)^{k(N^2-1)/8}(-1)^{(N-1)(b-1)/4}\left(\frac{N}{b}\right) \\ &= (-1)^{k(N^2-1)/8+(N-1)(b-1)/4}\left(\frac{M}{b}\right).\end{aligned}$$

The dominant cost of the algorithm is the remainder computation in step 3, which takes $O(\log N \cdot \log b)$ word operations with classical arithmetic. With the exception of step 2, the computations in the recursive process are essentially the same as in the Euclidean Algorithm, and a similar analysis as in Section 3.3 shows that the overall cost is $O(\log N \cdot \log a)$ word operations.

18.24 (i) This follows immediately from Lemma 14.7.

(ii) Let $N = p^e m$ for a prime $p \in \mathbb{N}$, $e \in \mathbb{N}_{\geq 1}$, and an integer $m > 1$ coprime to p. We may assume that $-1 \in T$, and let $c \in \mathbb{N}$ be such that $\sigma(c \bmod N) = -1$. Using the Chinese Remainder Theorem, we can find a $b \in \mathbb{N}$ such that $b \equiv c \bmod p^e$ and $b \equiv 1 \bmod m$. Then $\gcd(b,N) = 1$ and $b \bmod N \in \mathbb{Z}_N^\times$. Moreover, $b^{(N-1)/2} \equiv c^{(N-1)/2} \equiv -1 \bmod p^e$ and $b^{(N-1)/2} \equiv 1 \bmod m$, and we conclude that $b^{(N-1)/2} \not\equiv \pm 1 \bmod N$.

(iii) By Lemma 18.1, $a = (1 + p^{e-1}) \bmod N$ has multiplicative order p in $\mathbb{Z}_N^\times$. Now $(N-1)/2 = (p^e - 1)/2$ is coprime to p, and hence $\sigma(a)$ has order p as well, by Exercise 14.11 (ii). Since $p \geq 3$, we conclude that $\sigma(a) \neq \pm 1$.

We remark that $T = \pm 1 + p\mathbb{Z}_N$ if e is odd and $T = 1 + p\mathbb{Z}_N$ if e is even.

(iv) If $T = \{1\}$, then $a^{N-1} = \sigma(a)^2 = 1$ for all $a \in \mathbb{Z}_N^\times$.

(v) Suppose first that N is prime. Then $b_i = \pm 1$ and, by Lemma 14.7, each of the two possible values occurs with probability $1/2$, for all i. The algorithm incorrectly returns "probably composite" if and only if either all b_i are 1 or all b_i are -1. Each of the two events happens with probability 2^{-k}, and hence the correctness probability is $1 - 2^{1-k}$.

Now assume that N is composite. If $\gcd(a_i,N) > 1$ for some i, then $\gcd(b_i,N) > 1$ as well and $b_i \neq \pm 1$, and the algorithm correctly returns "probably composite". Thus it is sufficient to show the probability estimate for the case where

$\gcd(a_i, N) = 1$ for all i. If $T = \{1\}$, then $b_i = 1$ for all i, and the algorithm correctly outputs "probably composite". Otherwise, (ii) and (iii) imply $G = T \cap \{\pm 1\}$ is a proper subgroup of T, so that $\#G \leq \#T/2$, by Lagrange's theorem. Since σ is a group homomorphism, Lagrange's theorem also implies that $\#\sigma^{-1}(G) \leq \#\mathbb{Z}_N^\times/2$, and we have $\sigma(a) \neq \pm 1$ for at least half of the elements in $\mathbb{Z}_N^\times$. Thus each b_i is different from ± 1 with probability at least $1/2$, for all i, and the probability that $b_i = \pm 1$ for all i is at most $1 - 2^{-k}$.

In fact, if $-1 \notin T$, then the algorithm *always* returns the correct result "probably composite".

(vi) Each execution of step 2 takes $O(\log N \cdot \mathsf{M}(\log N))$ word operations, and this dominates the cost for the other steps. There are k iterations of the loop 1, and the claim follows.

(vii) If N is composite, then the same proof as in (v) shows that the modified algorithm returns the correct answer with probability at least $1 - 2^{-k}$. If N is prime, then the correctness probability of the modified algorithm is $1 - 2^{-k}$ as well.

(viii) We have $343 = 7^3$, so $T_{343} = \pm 1 + 7\mathbb{Z}_{343}$ and $\#T_{343} = 2 \cdot 49 = 98$, by the remark at the end of (iii). $561 = 3 \cdot 11 \cdot 17$ is the smallest Carmichael number, and $T_{561} = \{1, 67\}$. For $N = 667 = 23 \cdot 29$, we have $T_{667} = \mathbb{Z}_{667}^\times$ and $\#T_{667} = \varphi(667) = 22 \cdot 28 = 616$, since $(667-1)/2 = 333 = 9 \cdot 37$ is coprime to the order $616 = 2^3 \cdot 7 \cdot 11$ of $\mathbb{Z}_{667}^\times$ and hence σ is an automorphism of $\mathbb{Z}_N^\times$. Finally, $841 = 29^2$, and $T_{841} = 1 + 29\mathbb{Z}_{841}$ and $\#T_{841} = 29$, again by the remark at the end of (iii).

For $N = 561$ and $N = 841$, we have $-1 \notin T$, and the algorithm returns "probably composite" in any case. For the other two numbers, the probability that $\sigma(a) = 1$ for a randomly chosen $a \in \mathbb{Z}_N^\times$ is $1/\#T$, and the probability that $\sigma(a) = -1$ is $1/\#T$ as well. The algorithm incorrectly returns "probably prime" if and only if $b_i = \pm 1$ for all i and the b_i are not all equal. This happens with probability $(2/\#T)^k - 2(1/\#T)^k$. If we let $k = 10$, then the exact error probability is $49^{-10} - 2 \cdot 98^{-10} < 1.251 \cdot 10^{-17}$ for $N = 343$, and $333^{-10} - 2 \cdot 666^{-10} \leq 5.953 \cdot 10^{-26}$ for $N = 667$. The estimate from (v) for the error probability is only $2^{-10} \approx 10^{-3}$.

18.25 (i) Let $n \geq 1$. Then $F_n \equiv 1 \bmod 4$. We have $F_n \equiv 2 \bmod 3$, so that F_n is not a square modulo 3, and the law of quadratic reciprocity implies that 3 is not a square modulo F_n. Similarly, $F_n \equiv 3 \bmod 7$ if n is even and $F_n \equiv 5 \bmod 7$ if n is odd, and both 3 and 5 are nonsquares modulo 7. Thus F_n is not a square modulo 7, and again the law of quadratic reciprocity implies that 7 is not a square modulo F_n. If $n \geq 2$, we have $F_n \equiv 2 \bmod 5$, so that F_n is not a square modulo 5, and hence 5 is not a square modulo F_n.

(ii) If F_n is prime, then 3 is not a square modulo F_n, and hence $3^{(F_n-1)/2} \equiv -1 \bmod F_n$, by Lemma 14.7. Conversely, suppose that $3^{(F_n-1)/2} \equiv -1 \bmod F_n$. Then $3^{F_n-1} \equiv 1 \bmod F_n$, and hence $m = \operatorname{ord}_{F_n}(3)$ divides $F_n - 1 = 2^{2^n}$. If $m \neq 2^{2^n}$, then $m = 2^k$ for some $k < 2^n$, so that $m \mid (F_n - 1)/2$, which is a contradiction to $3^{(F_n-1)/2} \equiv -1 \bmod F_n$. Thus $m = 2^{2^n}$, and Exercise 18.27 (ii) implies that F_n is prime.

(iii) Computing $3^{(F_n-1)/2}$ rem F_n by repeated squaring takes $O(\log F_n \cdot \mathsf{M}(\log F_n))$ or $O(2^n \cdot \mathsf{M}(2^n))$ word operations.

18.26 Let p be a prime divisor of F_n. Then $2^{2^n} = F_n - 1 \equiv -1 \bmod p$ and $2^{2^{n+1}} \equiv 1 \bmod p$, and Exercise 8.16 (i) implies that $\operatorname{ord}_p(2) = 2^{n+1}$. Lagrange's theorem implies that $8 \mid 2^{n+1} \mid p-1$, and hence $p^2 - 1 = (p+1)(p-1) \equiv 0 \bmod 16$. Then Exercise 18.23 (iii) shows that $(\frac{2}{p}) = 1$, and hence 2 is a square modulo p. If $a \in \mathbb{N}$ is such that $a^2 \equiv 2 \bmod p$, then $\operatorname{ord}_p(a) = 2^{n+2}$, and again Lagrange's theorem implies that $2^{n+2} \mid p-1$.

18.27 In part (iv) of this exercise in the 1999 edition, co-$\mathcal{NP}$ should be replaced by $\mathcal{NP} \cap$ co-$\mathcal{NP}$.

(ii) If N is prime, then Exercise 8.16 implies that N has a Pratt witness. Conversely, if $u \in \mathbb{Z}_N^\times$ has order $N-1$, then $N-1 \mid \mathbb{Z}_N^\times \leq N-1$, by Lagrange's theorem, and hence $\mathbb{Z}_N^\times = \mathbb{Z}_N \setminus \{0\}$ and N is prime.

(iii) The certificates are given in Figure 18.3; the certificates $(\mathbf{2}, 1)$ for 2 are omitted. The number of distinct certificates depends on the number of generators of the

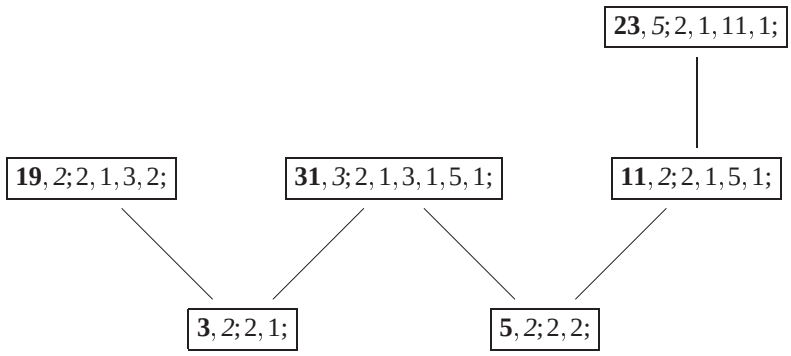

FIGURE 18.3: Various Pratt certificates (Exercise 18.27)

multiplicative group $\mathbb{Z}_N^\times$ for a prime N. More precisely, it is equal to the number of generators times the product of the number of certificates for all prime divisors of $N-1$. Let $u \in \mathbb{Z}_N^\times$ be such a generator. Then each generator is of the form u^k for some $k \in \{1, \ldots, N-2\}$ with $\gcd(k, N-1) = 1$, by Exercise 14.6 (ii). Thus the number of generators is $\varphi(N-1)$. We have $\varphi(3-1) = 1$, $\varphi(5-1) = 2$, $\varphi(11-1) = 4$, $\varphi(19-1) = 6$, $\varphi(23-1) = 10$, and $\varphi(31-1) = 8$. The following table gives the number of distinct Pratt certificates for the above numbers.

N	3	5	11	19	23	31
	1	2	8	6	80	16

(iv) Let $C = (N, u; p_1, e_1, \ldots, p_r, e_r; C_1, \ldots, C_r)$ be a Pratt certificate and $n = \log N$. The cost for checking whether $N = p_1^{e_1} \cdots p_r^{e_r}$ is $O(\mathsf{M}(n) \log n)$ word operations, using an integer analog of Algorithm 10.3. Computing u^{N-1} rem N and $u^{(N-1)/p_i}$ rem N for all i by repeated squaring takes $O(rn\mathsf{M}(n))$ or $O(n^2\mathsf{M}(n))$ word operations since $r \leq \log N$. If we arrange the certificate in form of a tree with C at the root and $C_1, \ldots, C_r$ children of C, then the sum of the binary lengths of the integers at one particular level is $O(n)$. Using the superlinearity of M, we find that the overall cost at each level is $O(n^2\mathsf{M}(n))$ word operations. Since $p_i < N/2$ for all i, the depth of the tree is $O(n)$, and we obtain a total cost of $O(n^3\mathsf{M}(n))$ word operations.

It follows that PRIMES$\in \mathcal{NP}$; PRIMES$\in$ co-$\mathcal{NP}$ is trivial: as certificate for a composite number N one can take a proper factor of N.

Chapter 19

19.2 We identify the elements of $\mathbb{F}_p$ with $0, \ldots, p-1$.

ALGORITHM 19.27 .
Input: A prime p and $f \in \mathbb{F}_p[x]$ monic of degree n and dividing $x^p - x$.
Output: All roots of f.

1. $k \longleftarrow \lceil \sqrt{p} \rceil, \quad S \longleftarrow \emptyset$
2. $g \longleftarrow \prod_{0 \leq i < k} (x - i)$
3. **for** $j = 0, \ldots, k-1$ compute $a_j \in \mathbb{F}_p[x]$ with $a_j \equiv g(x - jk) \bmod f$ and degree less than n
4. **for** $j = 0, \ldots, k-1$ **do**
   $h_j \longleftarrow \gcd(a_j, f)$
   if $h_j \neq 1$ **then** $S \longleftarrow S \cup \{i : jk \leq i < (j+1)k \text{ and } h_j(i) = 0\}$
5. **return** S

Correctness of the algorithm follows immediately from

$$h_j = \gcd(a_j, f) = \gcd(g(x - jk), f) = \prod_{\substack{jk \leq i < (j+1)k \\ f(i) = 0}} (x - i)$$

for all j. By Lemma 10.4, we can compute the coefficients of g in step 2 using $O(\mathsf{M}(p^{1/2}) \log p)$ arithmetic operations in $\mathbb{F}_p$. In step 3, we use the fast multipoint evaluation algorithm 10.7 over $R = \mathbb{F}_p[x]/\langle f \rangle$ to evaluate $g \in R[x]$ at the k points $x \bmod f, x - k \bmod f, \ldots, x - (k-1)k \bmod f$ in R, taking $O(\mathsf{M}(p^{1/2}) \log p)$ additions and multiplications in R or $O(\mathsf{M}(p^{1/2}) \log p \cdot \mathsf{M}(n))$ arithmetic operations in $\mathbb{F}_p$. The cost for computing h_j in step 4 is $O(\mathsf{M}(n) \log n)$ arithmetic

operations in $\mathbb{F}_p$, and $O(p^{1/2}\mathsf{M}(n)\log n)$ for all j. Evaluating h_j at k points in $\mathbb{F}_p$ takes $O(p^{1/2}\deg h_j)$ operations in $\mathbb{F}_p$. Since $\sum_{0\le j<k}\deg h_j \in O(n)$, the overall cost for the evaluations is $O(p^{1/2}n)$ operations in $\mathbb{F}_p$. Thus the total cost is $O(\mathsf{M}(p^{1/2})\log p\cdot\mathsf{M}(n)+p^{1/2}\mathsf{M}(n)\log n)$ or $O^{\sim}(n\sqrt{p})$ arithmetic operations in $\mathbb{F}_p$.

19.3 $N = 12347\cdot 1927836461$.

19.5 (i) Here is a table of the integers x_i modulo the primes $p \le 11$ for $i \le 6$.

i	$x_i \bmod 2$	$x_i \bmod 3$	$x_i \bmod 5$	$x_i \bmod 7$	$x_i \bmod 11$
0	0	2	2	2	2
1	1	2	0	5	5
2	0	2	1	5	4
3	1	2	2	5	5
4	0	2	0	5	4
5	1	2	1	5	5
6	0	2	2	5	4

We read off $e(2)=2$, $e(3)=1$, $e(5)=3$, $e(7)=1$, and $e(11)=2$.

(iii) Let p be a prime divisor of N and $i=e(p)>0$. Then p divides $x_i - x_{2i}$, and hence also $\gcd(x_i-x_{2i},N)$. Since the latter gcd is 1 for $i\le k$, we find $i>k$.

(iv) follows from (ii) and (iii).

19.6

$$\begin{aligned}\left(\int_{-\infty}^{\infty} e^{-x^2}dx\right)^2 &= \int_{-\infty}^{\infty} e^{-x^2}dx\cdot\int_{-\infty}^{\infty} e^{-y^2}dy = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-x^2-y^2}dx\,dy\\ &= \int_0^{\infty}\int_{-\pi}^{\pi} e^{-r^2(\cos^2\varphi+\sin^2\varphi)}r\,d\varphi\,dr = \int_0^{\infty} re^{-r^2}\int_{-\pi}^{\pi} d\varphi\,dr\\ &= 2\pi\int_0^{\infty} re^{-r^2}dr = -\pi\cdot\left(e^{-r^2}\Big|_{r=\infty} - e^{-r^2}\Big|_{r=0}\right) = \pi,\end{aligned}$$

where we have used that the absolute value of the Jacobi determinant of the substitution $(x,y)=f(r,\varphi)=(r\cos\varphi, r\sin\varphi)$ for $r\ge 0$ is

$$\left|\det\begin{pmatrix}\dfrac{\partial f(r,\varphi)}{\partial r}\\[2mm] \dfrac{\partial f(r,\varphi)}{\partial\varphi}\end{pmatrix}\right| = \left|\det\begin{pmatrix}\cos\varphi & \sin\varphi\\ -r\sin\varphi & r\cos\varphi\end{pmatrix}\right| = r.$$

19.8 (i) Let $(x_*,y_*,z_*)\in\mathbb{N}^3_{\ge 1}$ be a Pythagorean triple, $\lambda=\gcd(x_*,y_*,z_*)$, and let $(x_*,y_*,z_*)=(\lambda x,\lambda y,\lambda z)$. Then $\gcd(x,y,z)=1$ and $x^2+y^2=(x_*^2+y_*^2)/\lambda^2 = z_*^2/\lambda^2 = z^2$.

(ii) We check that

$$(s^2-t^2)^2+(2st)^2 = s^4-2s^2t^2+t^4+4s^2t^2 = s^4+2s^2t^2+t^4 = (s^2+t^2)^2.$$

Let $\lambda = \gcd(s^2-t^2, 2st, s^2+t^2)$. Then $\lambda \mid 2s^2$ and $\lambda \mid 2t^2$, and $\gcd(s,t) = 1$ implies that $\lambda \mid 2$. Since st is even, exactly one of s and t is even, and hence s^2+t^2 is odd. Thus $\lambda = 1$.

(iii) If both x and y were even, then both $z^2 = x^2+y^2$ and z would be even as well, contradicting primitivity. If both x and y were odd, then $z^2 \equiv 2 \bmod 4$, which is impossible since 0 and 1 are the only squares modulo 4. Thus exactly one of x and y is even, and both $z^2 = x^2+y^2$ and z are odd. If x is odd, then both $z-x$ and $z+x$ are even, and hence $(z-x)/2$ and $(z+x)/2$ are positive integers. If $\lambda \in \mathbb{N}$ is a common divisor of these two numbers, then it divides their sum z, their difference $-x$, and their product $(y/2)^2$, and hence $\lambda = 1$ since x,y,z are coprime and x,z are odd. But the product of two coprime numbers is only a square when both numbers are themselves squares, and hence there exist positive coprime $s,t \in \mathbb{N}$ with $s^2 = (z+x)/2$ and $t^2 = (z-x)/2$. Then $s > t$, s is odd and t is even if $z \equiv x \bmod 4$, and s is even and t is odd if $z \equiv -x \bmod 4$. Thus st is even, and one easily checks that $(x,y,z) = (s^2-t^2, 2st, s^2+t^2)$.

(iv) Here is a list of all coprime pairs $s,t \in \mathbb{N}_{\geq 1}$ with $s > t$ and st even and such that $s^2+t^2 \leq 100$, and the corresponding primitive Pythagorean triples $(x,y,z) = (s^2-t^2, 2st, s^2+t^2)$.

s	t	(x,y,z)
2	1	(3,4,5)
3	2	(5,12,13)
4	1	(15,8,17)
4	3	(7,24,25)
5	2	(21,20,29)
5	4	(9,40,41)
6	1	(35,12,37)
6	5	(11,60,61)

s	t	(x,y,z)
7	2	(45,28,53)
7	4	(33,56,65)
7	6	(13,84,85)
8	1	(63,16,65)
8	3	(55,48,73)
8	5	(39,80,89)
9	2	(77,36,85)
9	4	(65,72,97)

19.9 We have $O^\sim(n) \subseteq n^{1+o(1)}$, but not vice versa: for example, $n^{1+(\log n)^{-1/2}} = ne^{\sqrt{\log n}}$ does not belong to $O^\sim(n)$. More precisely, $O^\sim(n) = n^{1+O(\log\log n/\log n)}$.

19.10 Let $J \subseteq I$ be the set of i with $\#(B_i \cap C) \geq sk/2$, and $l = \#J$. Then

$$\begin{aligned} lk + \frac{s\#A}{2} = lk + \frac{sk\#I}{2} &\geq lk + (\#I - l)\left\lfloor \frac{sk-1}{2} \right\rfloor \\ &\geq \sum_{i\in J} \#(B_i \cap C) + \sum_{i \in I\setminus J} \#(B_i \cap C) = \#C = s\#A, \\ l &\geq \frac{s\#A}{2k} = \frac{s\#I}{2}. \end{aligned}$$

19.14 (i) We have $f' = 3x^2 + a$ and

$$\operatorname{res}(f, f') = \det \begin{pmatrix} 1 & 0 & 3 & 0 & 0 \\ 0 & 1 & 0 & 3 & 0 \\ a & 0 & a & 0 & 3 \\ b & a & 0 & a & 0 \\ 0 & b & 0 & 0 & a \end{pmatrix} = 4a^3 + 27b^2.$$

(ii) This follows from the fact that f is squarefree if and only if it is coprime to its derivative and Corollary 6.17.

(iii) For $a = -1$, we have $r = -4 + 27b^2 = 0$ if and only if $b = \pm 2\sqrt{3}/9 \approx \pm 0.3849$.

19.15 The equation of the tangent at $P = (x_1, y_1)$ is given by the first order Taylor expansion of $f = y^2 - x^3 - ax - b$ around P:

$$\begin{aligned} t(x,y) &= f(x_1,y_1) + \frac{\partial f}{\partial x}(x_1,y_1) \cdot (x - x_1) + \frac{\partial f}{\partial y}(x_1,y_1) \cdot (y - y_1) \\ &= -(3x_1^2 + a)(x - x_1) + 2y_1(y - y_1), \end{aligned}$$

and the tangent is defined by the equation $t = 0$. Since E is nonsingular, t is not the zero polynomial, and at least one of $3x_1^2 + a$ and y_1 is nonzero. If $y_1 = 0$, then the equation of the tangent is $x = x_1$. This is a vertical line through P, and its only other intersection point with the curve is the point $\mathcal{O}$ at infinity. Thus $P + P = -\mathcal{O} = \mathcal{O}$ in that case.

Otherwise, if $y_1 \neq 0$, then we can solve $t = 0$ for y and obtain the equivalent equation

$$y = y_1 + \frac{3x_1^2 + a}{2y_1}(x - x_1) = y_1 + \alpha(x - x_1). \tag{17}$$

This is the unique line through P with slope $\alpha = (3x_1^2 + a)/2y_1$. To find the only other intersection point $S = (x_3, -y_3)$ of the tangent with the curve, we replace x, y by $x_3, -y_3$ in (17) and substitute the expression for y_3 that we obtain into the equation $f(x_3, y_3) = 0$:

$$(\alpha x_3 + y_1 - \alpha x_1)^2 = y_3^2 = x_3^2 + ax_3 + b.$$

Now $g = u^3 + au + b - (\alpha u + y_1 - \alpha x_1)^2$ is a cubic polynomial in u which has $u = x_1$ as a root. Differentiating with respect to u yields

$$g' = 3u^2 + a - 2\alpha(\alpha u + y_1 - \alpha x_1),$$

and we see that $g'(x_1) = 0$ and $u = x_1$ is in fact a double root of g (this mirrors the geometric situation that P is a double point of intersection of the tangent with the curve). The third root of g is x_3, the coordinate we are interested in, and hence $g = (u - x_1)^2(u - x_3)$. Thus the coefficient of u^2 in g is equal to $-2x_1 - x_3$, and solving this equality for x_3 yields $x_3 = -\alpha^2 - 2x_1$, as in (11). Finally, we plug $S = (x_3, -y_3)$ into the equation (17) for the tangent and obtain $-y_3 = y_1 + \alpha(x_3 - x_1)$.

19.16 $P=(x_1,y_1)$ has order two if and only if and $P+P=\mathcal{O}$, or equivalently, $y_1=0$. But the equation $y_1^2=x_1^3+ax_1+b$ has at most three roots x_1, and the claim follows. The solutions correspond to the three points of E on the x-axis.

19.17 The following MAPLE program is the main step in proving the associative law for addition on elliptic curves. Addition is given by the procedure `add`. Associativity is tested at the three generic inputs P,Q,R. They are given by their $3\cdot 2=6$ coordinates, but should correspond to only five free choices (two for the curve, one each for the points), and we have to compute modulo the equation `eq`$=v^2-(u^3+au+b)$, whose parameters a and b we find by elimination from P and Q.

```
plus := proc(P,Q)   # adding P=(P[1], P[2]) and Q=(Q[1], Q[2])
s := (P[2] - Q[2]) / (P[1] - Q[1]);
x3 := normal(s^2 - P[1] - Q[1]);
[x3, normal(s * (P[1] - x3) - P[2])];
end;

P := [x1, x2];
Q := [y1, y2];
R := [z1, z2];
a := (y1^2 - y2^2)/(x1 - x2) - (x1^2 + x1 * x2 + x2^2);
b := y1^2 - x1^3 - a * x1;
eq := numer(normal(z1^3 + a * z1 + b - z2^2));
ass1 := plus(plus(P, Q), R);
ass2 := plus(P, plus(Q, R));
# The following are the differences of the two coordinates
# of (P + Q) + R and P + (Q + R), and hopefully turn out
# to be zero.
zero1 := rem(numer(normal(ass1[1] - ass2[1])), eq, y3);
zero2 := rem(numer(normal(ass1[2] - ass2[2])), eq, y3);
```

19.18 (i) By using the symmetry $\binom{2n}{k}=\binom{2n}{2n-k}$ of the binomial coefficients, we find that

$$2^{2n}=\sum_{0\le k\le 2n}\binom{2n}{k}=\binom{2n}{n}+2\sum_{0\le k<n}\binom{2n}{k}.$$

The other formula is proven similarly.

(ii) We have $X=2(n-k)$ if and only if $X_i=-1$ for precisely k values of i. Of the 2^{2n} equally probable random vectors $(X_1,\ldots,X_{2n})\in\{1,-1\}^{2n}$, exactly $\binom{2n}{k}$ satisfy this requirement, and hence the probability is $\binom{2n}{k}/2^{2n}$. The argument for $X=-2(n-k)$ follows by symmetry considerations.

(iii) It is clear that X can only take even values between $-2n$ and $2n$. For $0 \le k \le n$, let p_k and q_k denote the probabilities of the events $X = 2(n-k)$ and $X = -2(n-k)$, respectively. Then (ii) shows that $p_k = q_k = \binom{2n}{k}4^{-n}$ for all k, and hence

$$\begin{aligned}
\mathcal{E}(X) &= \sum_{k\in\mathbb{Z}} k\cdot\mathrm{prob}(X=k) = \sum_{0\le k<n}\Big(2(n-k)p_k - 2(n-k)q_k\Big) = 0,\\
\mathcal{E}(|X|) &= \sum_{k\in\mathbb{N}} k\cdot\mathrm{prob}(|X|=k) = \sum_{0\le k<n}\Big(2(n-k)p_k + 2(n-k)q_k\Big)\\
&= 4\cdot 4^{-n}\sum_{0\le k<n}(n-k)\binom{2n}{k}\\
&= 4^{1-n}\left(n\sum_{0\le k<n}\binom{2n}{k} - 2n\sum_{1\le k<n}\binom{2n-1}{k-1}\right)\\
&= n4^{1-n}\left(\frac{1}{2}\Big(4^n - \binom{2n}{n}\Big) - 2\Big(4^{n-1} - \binom{2n-1}{n-1}\Big)\right)\\
&= n4^{1-n}\left(2\binom{2n-1}{n-1} - \frac{1}{2}\binom{2n}{n}\right) = 2n4^{-n}\binom{2n}{n}.
\end{aligned}$$

(iv) We have

$$\begin{aligned}
2n4^{-n}\binom{2n}{n} &= 2n4^{-n}\frac{(2n)!}{(n!)^2} \in 2n4^{-n}\frac{\sqrt{4\pi n}(2n)^{2n}e^{-2n}(1+O(n^{-1}))}{2\pi n\cdot n^{2n}e^{-2n}(1+O(n^{-1}))}\\
&= \frac{4\sqrt{\pi}n^{3/2}}{2\pi n}(1+O(n^{-1})) = 2\pi^{-1/2}n^{1/2}(1+O(n^{-1})),
\end{aligned}$$

where we used that $1/(1+O(n^{-1})) \in (1+O(n^{-1}))$ for $n \longrightarrow \infty$.

Chapter 20

20.1 (i) The cleartext is "COMPUTER", and the key is $k = 12$.
(ii) "ALGEBRA".

20.3 Part (iii) needs the additional assumption that r is coprime to $\mathrm{char}\,F$.
(i) We have

$$g(cx+d)\circ\frac{h-d}{c} = g\Big(c\frac{h-d}{c}+d\Big) = g(h) = f.$$

If we choose $d = h(0)$, $c = \mathrm{lc}(h)$, $g^* = g(cx+d)/\mathrm{lc}(g(cx+d))$, and $h^* = (h-d)/c$, then g^*, h^* are monic, $h^*(0) = 0$, and $f = \mathrm{lc}(f)\cdot g^*(h^*)$.
(ii) Let $g = x^r + \sum_{0\le i<r} g_i x^i$, with all $g_i \in F$. Then $f = h^r + \sum_{0\le i<r} g_i h^i$. Substituting $x = 1/x$ and multiplying by x^{rs}, we find

$$\begin{aligned}
f^* &= x^{rs}f(1/x) = x^{rs}h(1/x)^r + \sum_{0\le i<r} g_i x^{rs}h(1/x)^i\\
&= (h^*)^r + x^s\sum_{0\le i<r} g_i x^{(r-1-i)s}(h^*)^i \equiv (h^*)^r \bmod x^s.
\end{aligned}$$

(iii) If we let $\varphi = y^r - f^* \in F[x][y]$, then $\varphi(h^*) = (h^*)^r - f^* \equiv 0 \equiv (h_1^*)^r - f^* = \varphi(h_1^*) \bmod x^s$, by (ii). Since both h and h_1 are monic, we have $h^* \equiv h_1^* \equiv 1 \bmod x$, and hence $\varphi'(h^*) \equiv \varphi'(h_1^*) \equiv r \bmod x$. Since we assume that r is coprime to $\operatorname{char} F$, both $\varphi'(h^*)$ and $\varphi'(h_1^*)$ are invertible modulo x. Thus $h^* \equiv h_1^* \bmod x^s$, by the uniqueness of Newton iteration (Theorem 9.27). Since $h(0) = h_1(0) = 0$, we conclude that $h = h_1$. Then $g(h) = g = g_1(h_1) = g_1(h)$, or equivalently, $(g - g_1)(h) = 0$. If $g \neq g_1$, then $\deg((g - g_1) \circ h) = \deg(g - g_1) \cdot \deg h \geq 0$, and hence $g = g_1$.

The case where $\operatorname{char} F$ divides r is more difficult; see von zur Gathen (1990b) for a discussion.

(iv) It is clear that the output is correct if the algorithm returns g and h in step 3. Conversely, we let $f = g_1 \circ h_1$ be a normal decomposition with $\deg g_1 = r$ and $\deg h_1 = s$. Then $(h_1^*)^r \equiv f^* \bmod x^s$, by (ii), and as in (iii), the uniqueness of Newton iteration implies that $h^* \equiv h_1^* \bmod x^s$ and $h_1 = h$ in step 2. Let $g_1 = x^r + \sum_{0 \leq i < r} g_{1i} x^i$. Then $f = g_1 \circ h$ implies that $f = h^r + \sum_{0 \leq i < r} g_{1i} h^i$. Now the h-adic expansion of f is unique, by Lemma 5.30, and hence $g_i = g_{1i} \in R$ for all i, and the algorithm correctly returns $g = g_1$ and $h = h_1$ in step 3.

Theorem 9.25 states that h^* can be computed from f^* with $O(r \cdot \mathsf{M}(n))$ arithmetic operations in R. The factor r in the estimate comes from the cost for evaluating the function φ from (iii), which has degree r in y, and its derivative $\varphi' = \partial\varphi/\partial y$ at $y = h^* \operatorname{rem} x^i$ for several $i \leq s$ in a Horner-like way. Due to the special structure of φ, we can do this much faster with repeated squaring (we have discussed this in the integer case in Section 9.5), and then the cost for step 2 is only $O(\mathsf{M}(n) \log r)$ arithmetic operations. The same estimate is valid for step 3, by Theorem 9.15.

If $\gcd(r, \operatorname{char} R) > 1$, then $\varphi'(1) = r$ is not a unit in R, and the Newton iteration does not work since 1 is not a proper starting solution. However, this does not imply that no normal decomposition exists. For example, if $R = \mathbb{F}_2$, then $x^4 + x^2 = (x^2 + x) \circ x^2$ is a normal decomposition.

(v) $f = (x^3 - x^2 + 2) \circ (x^2 + 2x + 2)$.

20.4 (i) $d = 5$, (ii) $x = 1999$.

20.5 (i) The claim is clear if $x = 0$. So we let $x > 0$ and assume that $p \mid x$. Since $x < N$, we then have $q \nmid x$. Moreover, $x^{de} \equiv x \equiv 0 \bmod p$ and $x^{de} \equiv x \bmod q$ since $(q-1) \mid \varphi(N) \mid de - 1$. Thus $x^{de} \equiv x \bmod N$, by the Chinese Remainder Theorem.

(ii) Assume again that $p \mid x$ and $q \nmid x$. Then $p \mid x^e$ and $q \nmid x^e$, and hence $p = \gcd(\varepsilon(x), N)$.

20.6 (i) We have $(x-p)(x-q) = x^2 - (p+q)x + N = x^2 - (N - 1 - \varphi(N))x + N$. Thus p and q can be found by solving a quadratic equation, for example, by using Algorithm 14.17 or Theorem 15.21, at a cost of $O^{\sim}(\log N)$ word operations.

(ii) We call the black box for $e = 2, 3, \ldots$. Since the product of all primes below $x = 2 \ln N$ is $e^{\vartheta(x)} > N$ if $x \geq 5$, by the solution of Exercise 18.21, there exists a prime $e \leq x$ with $\gcd(e, \varphi(N)) = 1$. Then the black box returns $d < \varphi(N)$ such that $\varphi(N) \mid ed - 1$. Now $ed - 1 < e\varphi(N)$, and we successively divide $ed - 1$ by

$1,2,\ldots,e-1$ and use the trial values for $\varphi(N)$ from those divisions that have the remainder zero to find p and q as in (i). The overall cost is $O(\log N)$ calls to the black box and $O(\log N)$ calls to the square root finding algorithm.

An alternative is to use the algorithm from Exercise 18.12 (ii) with $L = ed - 1$. (The number L was called m in the 1999 edition.)

Chapter 21

21.1 If $F[x,y]$ were Euclidean, then we could use the EEA to compute $s,t \in F[x,y]$ such that $sx+ty = 1 = \gcd(x,y)$. Now substituting $x = y = 0$ leads to the contradiction $0 = 1$.

21.2 It is clear that $I \subseteq J$. For the reverse inclusion, we note that

$$x = (1-y)\cdot(x+xy)+x\cdot y^2 \in I \text{ and } y = (1-x)\cdot(y+xy)+y\cdot x^2 \in I.$$

21.7 If $\alpha < \beta$ and $\beta < \alpha$, then the transitivity of $<$ implies $\alpha < \alpha$, contradicting irreflexivity.

21.8 Suppose that $\alpha \prec 0$ for some $\alpha \in \mathbb{N}^n \setminus \{0\}$. Adding $i\alpha$ to both sides of the inequality, we find $(i+1)\alpha \prec i\alpha$ for all $i \in \mathbb{N}$. Thus $\{i\alpha : i \in \mathbb{N}\}$ is a nonempty subset of $\mathbb{N}^n$ with no least element, contradicting the well-order property. Since $\prec$ is a total order, we conclude that $\alpha \succ 0$.

21.10 Let $x_1^{d_1}\cdots x_n^{d_n}$ be a monomial of total degree $m = d_1+\cdots+d_n$. We associate to it the vector

$$v = (\underbrace{0,\ldots,0}_{d_1},1,\underbrace{0,\ldots,0}_{d_2},1,\ldots,1,\underbrace{0,\ldots,0}_{d_n}) \in \{0,1\}^{m+n-1}.$$

This induces a bijection between the monomials of total degree m and the binary vectors of length $m+n-1$ with precisely $n-1$ ones, and the number of the latter is $\binom{m+n-1}{n-1} = \binom{m+n-1}{m}$.

21.11 We say that a monomial $\boldsymbol{x}^\alpha$ occurs in a polynomial h if its coefficient in h is nonzero.

(i) Every monomial occurring in fg is of the form $\boldsymbol{x}^{\alpha+\beta}$ such that $\boldsymbol{x}^\alpha$ and $\boldsymbol{x}^\beta$ occur in f and g, respectively. This implies that $\operatorname{mdeg}(fg) \preceq \operatorname{mdeg}(f)+\operatorname{mdeg}(g)$. On the other hand, if either $\alpha \prec \operatorname{mdeg}(f)$ or $\beta \prec \operatorname{mdeg}(g)$, then $\alpha+\beta \prec \operatorname{mdeg}(f)+\operatorname{mdeg}(g)$. Thus the coefficient of $\boldsymbol{x}^{\operatorname{mdeg}(f)+\operatorname{mdeg}(g)}$ is $\operatorname{lc}(f)\operatorname{lc}(g) \neq 0$, and hence $\operatorname{mdeg}(fg) = \operatorname{mdeg}(f)+\operatorname{mdeg}(g)$.

(ii) Every nonzero term of $f+g$ is of the form $(c+d)\boldsymbol{x}^\alpha$, for a coefficient c of f and a coefficient d of g. Since $c+d \neq 0$, at least one of c and d is nonzero, which implies that $\alpha \preceq \operatorname{mdeg}(f)$ or $\alpha \preceq \operatorname{mdeg}(g)$, and hence $\operatorname{mdeg}(f+g) \preceq \max\{\operatorname{mdeg}(f),\operatorname{mdeg}(g)\}$. If $\operatorname{mdeg}(f) \prec \operatorname{mdeg}(g)$, then the coefficient of $\boldsymbol{x}^{\operatorname{mdeg}(g)}$ in $f+g$ is $\operatorname{lc}(g)$, and hence $\operatorname{mdeg}(f+g) = \operatorname{mdeg}(g) = \max\{\operatorname{mdeg}(f),\operatorname{mdeg}(g)\}$. The claim for $\operatorname{mdeg}(f) \succ \operatorname{mdeg}(g)$ follows by a symmetric argument.

21.12 It is clear that the invariants hold after step 1. We now assume that they hold at the beginning of step 3 and show that they hold again at the end of step 3. We denote the new values of $p, r, q_1, \ldots, q_s$ by $p^*, r^*, q_1^*, \ldots, q_s^*$. When the condition in step 3 is false, then

$$\begin{aligned} \operatorname{mdeg}(p^*) &= \operatorname{mdeg}(p - \operatorname{lt}(p)) \prec \operatorname{mdeg}(p) \preceq \operatorname{mdeg}(f), \\ p^* + r^* &= p - \operatorname{lt}(p) + r + \operatorname{lt}(p) = p + r, \end{aligned}$$

and since $q_i^* = q_i$ for all i, the first two invariants holds for the starred elements. By induction and since the condition in step 3 is false, the last invariant also holds for r^*.

Now we assume that the condition in step 3 is true. Then $r^* = r$, and the last invariant holds for r^* by induction. We have

$$\operatorname{mdeg}(p^*) = \operatorname{mdeg}\left(p - \frac{\operatorname{lt}(p)}{\operatorname{lt}(f_i)} f_i\right) \prec \operatorname{mdeg}(p) \preceq \operatorname{mdeg}(f),$$

since both polynomials in the difference have degree $\operatorname{mdeg}(p)$ and their leading coefficients coincide. Moreover,

$$p^* + q_i^* f_i = p - \frac{\operatorname{lt}(p)}{\operatorname{lt}(f_i)} f_i + \left(q_i + \frac{\operatorname{lt}(p)}{\operatorname{lt}(f_i)}\right) f_i = p + q_i f_i,$$

and since $r^* = r$ and $q_j^* = q_j$ for $j \neq i$, the first invariant holds for the starred elements. Finally, if $q_i = 0$ then

$$\operatorname{mdeg}(q_i^* f_i) = \operatorname{mdeg}\left(\frac{\operatorname{lt}(p)}{\operatorname{lt}(f_i)} f_i\right) = \operatorname{mdeg}(p) \preceq \operatorname{mdeg}(f),$$

and similarly

$$\operatorname{mdeg}(q_i^* f_i) = \operatorname{mdeg}\left(q_i f_i + \frac{\operatorname{lt}(p)}{\operatorname{lt}(f_i)} f_i\right) \preceq \max\{\operatorname{mdeg}(q_i f_i), \operatorname{mdeg}(p)\} \preceq \operatorname{mdeg}(f)$$

if both q_i, q_i^* are nonzero. Since $q_j^* = q_j$ for $j \neq i$, this proves that the second invariant holds for the starred elements.

21.13 $(\alpha_1 + 1) \cdots (\alpha_n + 1)$.

21.14 If E is any subset of A such that $\langle \boldsymbol{x}^E \rangle = \langle \boldsymbol{x}^A \rangle$, then $\boldsymbol{x}^\beta \in \langle \boldsymbol{x}^E \rangle$ for all $\beta \in B$. Thus $\beta \geq \alpha$ for some $\alpha \in E$, by Lemma 21.15, and the minimality of β implies that $\beta = \alpha \in E$, which proves that E contains B.

21.15 For example, $I = \langle \{x^i y^{n-1-i} : 0 \leq i < n\} \rangle$.

21.17 Let $t \in \mathrm{lt}(I)$ and $f \in I$ with $\mathrm{lt}(f) = t$. Since f rem $(f_1, \ldots, f_s) = 0$, there exist $q_1, \ldots, q_s \in F[x_1, \ldots, x_n]$ with $f = q_1 f_1 + \cdots + q_s f_s$ and $\mathrm{mdeg}(q_i f_i) \preceq \mathrm{mdeg}(f)$ if $q_i \neq 0$. But then

$$t = \mathrm{lt}(f) = \sum_{\mathrm{mdeg}(q_i f_i) = \mathrm{mdeg}(f)} \mathrm{lt}(q_i)\,\mathrm{lt}(f_i) \in \langle \mathrm{lt}(f_1), \ldots, \mathrm{lt}(f_s) \rangle,$$

and hence $\mathrm{lt}(I) \subseteq \langle \mathrm{lt}(f_1), \ldots, \mathrm{lt}(f_s) \rangle$. The reverse inclusion is trivial, and $f_1, \ldots, f_s$ is a Gröbner basis.

21.18 Let $G' = G \setminus \{g\}$. Since $\mathrm{lt}(g) \in \langle \mathrm{lt}(G') \rangle$, we have $\langle \mathrm{lt}(G') \rangle = \langle \mathrm{lt}(G) \rangle = \mathrm{lt}(I)$.

21.19 Let $R = F[x_1, \ldots, x_n]$. If G contains a nonzero constant, then $I = \langle G \rangle = R$ and $1 \in I$. Conversely, if $1 \in I$, then $1 = \mathrm{lt}(1) \in \mathrm{lt}(I) = \langle \mathrm{lt}(G) \rangle$. By Lemma 21.15, there exists a $g \in G$ with $\mathrm{lt}(g) \mid 1$, and hence g is a nonzero constant. If now G is reduced, then $g = 1$ since g is monic. If G contains another polynomial $g^* \neq g$, then $1 \mid \mathrm{lt}(g^*)$ contradicts the minimality of G, and hence $G = \{1\}$.

21.20 (i) The reduced Gröbner basis is $G = \{x^2 + y - 1, xy - x, y^2 - 2y + 1\}$.

(ii) We have f_1 rem $G = 0$ and f_2 rem $G = 1$, and hence $f_1 \in I$ and $f_2 \notin I$.

21.21 (i) This is a reduced Gröbner basis since $S(x + y, y^2 - 1) = y^3 + x$ and $y^3 + x$ rem $(x + y, y^2 - 1) = 0$.

(ii) This is not a Gröbner basis since the leading term xy of $S(y + x, y^2 - 1) = xy + 1$ is neither divisible by $y = \mathrm{lt}(y + x)$ nor by $y^2 = \mathrm{lt}(y^2 - 1)$.

(iii) This is a not Gröbner basis; the reduced Gröbner basis for the generated ideal is $\{1\}$.

(iv) This is a Gröbner basis, but not a minimal one: $xyz = \mathrm{lt}(xyz - 1)$ is divisible by $x = \mathrm{lt}(x - y)$.

21.24 Let $R = F[x_1, \ldots, x_n]$.

(i) The polynomials in G_{LA} are F-linear combinations of the polynomials in G_A, which shows that $I_{LA} \subseteq I_A$; this holds for any $n \times n$ matrix L. If L is invertible, then the above argument shows that $I_A = I_{L^{-1}LA} \subseteq I_{LA}$.

(ii) By (i), we have $\langle G_U \rangle = I_U = I_A$, and it remains to show that G_U is a reduced Gröbner basis. Let $g_i = x_i + h_i$ correspond to the ith row of U, such that h_i is an F-linear combination of $x_{r+1}, \ldots, x_n$, for $1 \leq i \leq r$. Then

$$S(g_i, g_j) = x_j g_i - x_i g_j = x_j h_i - x_i h_j = g_j h_i - g_i h_j$$

and $S(g_i, g_j)$ rem $(g_1, \ldots, g_r) = 0$ for $i \neq j$. Thus G_U is a Gröbner basis. Now g_i is monic, and since x_i does not occur in g_j, we find that $x_i = \mathrm{lt}(g_i)$ does not divide any term in g_j, for $j \neq i$. Thus G_U is reduced.

(iii) If A is nonsingular, then $V(I_A) = \ker A = \{0\}$ and $G = \{x_1, \ldots, x_n\}$ is the reduced Gröbner basis of I_A.

21.26 The polynomial x^2-2 has no root in $\mathbb{F}_5$. Thus the ideal $I=\langle x^2-2\rangle \subsetneq \mathbb{F}_5[x]$ has no root in $\mathbb{F}_5$.

21.27 Let $I \subseteq \mathbb{C}[x]$ be an ideal and $g \in \mathbb{C}[x]$ such that $g(u)=0$ for all $u \in V(I)$. If $I=\{0\}$, then $V(I)=\mathbb{C}$, and this implies that $g=0$. Thus we may assume that $I \neq \{0\}$. Since $\mathbb{C}[x]$ is a Euclidean domain, there is a unique nonconstant monic polynomial $f \in \mathbb{C}[x]$ generating I. If $f=1$, then $I=\mathbb{C}[x]$, $V(I)=\emptyset$, and trivially $g \in I$. Now we assume that f is nonconstant. By the fundamental theorem of algebra, f splits into linear factors. Let $u_1,\ldots,u_r \in \mathbb{C}$ and $e_1,\ldots,e_r \in \mathbb{N}_{\geq 1}$ be such that $f=\prod_{1\leq i\leq r}(x-u_i)^{e_i}$. Then $V(I)=\{u_1,\ldots,u_r\}$. Since $g(u_i)=0$ for $1\leq i\leq r$, we conclude that $(x-u_1)\cdots(x-u_r) \mid g$. But then $f \mid g^e$ for $e=\max\{e_i : 1\leq i\leq r\}$, and hence $g^e \in \langle f\rangle = I$.

Chapter 22

22.2 (i) $D(1)=D(1\cdot 1)=D(1)\cdot 1+1\cdot D(1)=2D(1)$, by the Leibniz rule, and subtracting $D(1)$ on both sides yields the claim.

(ii) We have $D(af)=D(a)f+aD(f)=aD(f)$ and $D(bg)=D(b)g+bD(g)=bD(g)$, and hence $D(af+bg)=D(af)+D(bg)=aD(f)+bD(g)$.

(iii) We first note that $0=D(1)=D(gg^{-1})=D(g)g^{-1}+gD(g^{-1})$, and hence $D(g^{-1})=-D(g)g^{-2}$. Thus

$$D(fg^{-1})=D(f)g^{-1}+fD(g^{-1})=(D(f)g-fD(g))g^{-2}.$$

(iv) We use induction on n. The case $n=0$ follows from (i), and if $n\geq 1$, then

$$\begin{aligned} D(f^n) &= D(ff^{n-1})=D(f)f^{n-1}+fD(f^{n-1})=D(f)f^{n-1}+(n-1)D(f)f^{n-1} \\ &= nD(f)f^{n-1}. \end{aligned}$$

(v) This follows immediately from the Leibniz rule.

22.4 Since the field of constants of $\mathbb{Q}(x)$ is a subfield, by Exercise 22.1, and contains 1, by Lemma 22.2 (i), it contains $\mathbb{Q}$, the subfield of $\mathbb{Q}(x)$ generated by 1. Now let $f=\sum_{0\leq i\leq n} f_i x^i \in \mathbb{Q}[x]$ of degree $n\geq 1$, with all $f_i \in \mathbb{Q}$. Then Lemma 22.2 implies that $f'=\sum_{0\leq i\leq n} i f_i x^{i-1}$. In particular, since $nf_n\neq 0$, we have $\deg f'=n-1$, and f' is not the zero polynomial. Now let $f=g/h$, with nonzero coprime polynomials $g,h\in\mathbb{Q}[x]$, such that $f'=0$. Using the quotient rule (Lemma 22.2 (iii)), we obtain $0=(g'h-h'g)/h^2$, and hence $g'h=h'g$. Since g and h are coprime, we have $h \mid h'$. If $h\notin\mathbb{Q}$, then $\deg h' < \deg h$, by the above. This contradiction shows that $h\in\mathbb{Q}$ and $g'h=h'g=0$, and we conclude that $g'=0$ and $g\in\mathbb{Q}$ as well.

22.5 In the 1999 edition, the text of the exercise contains some errors, and we first give a corrected version of it.

Let F be a field of characteristic zero, and $a,b,c,d\in F[x]$ nonzero polynomials such that $(c/d)'=a/b$.

(i) Prove that $\deg a - \deg b \leq \deg c - \deg d - 1$, with equality if and only if $\deg c \neq \deg d$. Give an example where equality does not hold. Conclude that $\deg a - \deg b = -1$ is impossible.

(ii) Let $p \in F[x]$ be irreducible and $v_p(a) = e \in \mathbb{N}$ if $p^e \mid a$ and $p^{e+1} \nmid a$ (this is the negative logarithm of the p-adic value of a, as in Example 9.31 (iii)), and similarly $v_p(b)$, $v_p(c)$, $v_p(d)$. Prove that $v_p(a) - v_p(b) \geq v_p(c) - v_p(d) - 1$, with equality if and only if $v_p(c) \neq v_p(d)$. Give an example where equality does not hold. Conclude that $v_p(a) - v_p(b) = -1$ is impossible, and that $v_p(b) \geq 2$ for every irreducible divisor of b if $\gcd(a,b) = 1$. In particular, b is not squarefree if it is nonconstant and coprime to a.

Solution:

(i) Using the quotient rule (Lemma 22.2 (iii)), we find that $a/b = (c'd - cd')/d^2$. Now $\deg c' < \deg c$ and $\deg d' < \deg d$, and hence

$$\deg a - \deg b = \deg(c'd - cd') - 2\deg d < \deg c + \deg d - 2\deg d = \deg c - \deg d.$$

Let $n = \deg c$ and $m = \deg d$. The coefficient of x^{n+m-1} in $c'd$ is $n\operatorname{lc}(c)\operatorname{lc}(d)$, and the coefficient of x^{n+m-1} in cd' is $m\operatorname{lc}(c)\operatorname{lc}(d)$. Thus the coefficient of x^{n+m-1} in $c'd - cd'$ vanishes if and only if $n = m$. If $n \neq m$, then $\deg a - \deg b = n - m - 1 \neq -1$, and if $n = m$, then $\deg a - \deg b < n - m - 1 = -1$.

(ii) We show first that $v_p(u') \geq v_p(u) - 1$ for all nonconstant $u \in F[x]$, with equality if $v_p(u) \geq 1$. Let $u = p^e w$ with $e \in \mathbb{N}$ and $p \nmid w$. Then $u' = (ep'w + pw')p^{e-1}$, by Lemma 22.2. Thus $v_p(u') \geq e - 1 = v_p(u) - 1$. Since p is irreducible, $p' \neq 0$, and $\deg p' < \deg p$, we find that p does not divide p'. If $e \geq 1$, then p does not divide $ep'w$ since it is coprime to w, and hence $p \nmid (ep'g + pg')$ and $v_p(u') = e - 1$.

Now let $c = p^e u$ and $d = p^f w$, with $e, f \in \mathbb{N}$ and $u, w \in F[x]$ not divisible by p. Then $v_p(c) - v_p(d) = e - f$. As in (i), the quotient rule implies that

$$\begin{aligned} \frac{a}{b} = \frac{c'd - cd'}{d^2} &= \frac{(ep'u + pu')p^{e-1}p^f w - p^e u(fp'w + pw')p^{f-1}}{p^{2f}w^2} \\ &= \frac{\Big((e-f)p'uw + (u'w - uw')p\Big)p^{e+f-1}}{p^{2f}w^2}. \end{aligned}$$

Since $p \nmid w$, we find that $v_p(a) - v_p(b) \geq e + f - 1 - 2f = e - f - 1$. Moreover, since p does not divide p' and u either, it does not divide $p'uw$. Thus p divides $((e-f)p'uw + (u'w - uw')p)$ if and only if $e = f$. If $e \neq f$, then $v_p(a) - v_p(b) = e - f - 1 \neq -1$, and if $e = f$, then $v_p(a) - v_p(b) > e - f - 1 = -1$. Now suppose that b is nonconstant and coprime to a, and let $p \in F[x]$ an irreducible divisor of b. Then $v_p(a) = 0$ and $v_p(b) \geq 1$, so that $v_p(a) - v_p(b) \leq -1$. By the above, we have strict inequality, and hence $v_p(b) \geq 2$ and $p^2 \mid b$.

22.8 (i) By the Leibniz rule, we have

$$bd' = b \cdot \sum_{2 \leq j \leq m} (j-1) g_j' \frac{d}{g_j} = d \sum_{2 \leq j \leq m} (j-1) g_j' \frac{b}{g_j},$$

and the right hand side is a polynomial. Thus $d \mid bd'$. For $2 \le i \le m$, g_i divides all summands of $\sum_{2\le i\le m}(j-1)g_j'b/g_j$ with $j \ne i$, and it is coprime to $(i-1)g_i'b/g_i$. Thus $\gcd(bd'/d, b) = 1$. Using s and t as assumed, we find

$$\begin{aligned}\frac{h^*}{g^*} &= \frac{s(bd'/d)+tb}{db} = \frac{sbd'/d - s'b + (t+s')b}{db} = \frac{sd'-s'd}{d^2} + \frac{t+s'}{d} \\ &= \left(\frac{-s}{d}\right)' + \frac{t+s'}{d},\end{aligned}$$

and hence we can take $u = -s$ and $v = t + s'$.

(ii) Computing $d = \gcd(g, g')$, the squarefree part $g_1 b = g/d$ of g, and the polynomials $b = \gcd(g_1 b, d)$, $g_1 = g_1 b/b$, $g^* = g/g_1$, and bd'/d takes $O(\mathsf{M}(n)\log n)$ arithmetic operations. Computing the numerators h_1, h^* in the partial fraction decomposition can be done with $O(\mathsf{M}(n)\log n)$ operations as well, and the same estimate is valid for computing s, t, and $s + t'$. Thus the cost for one step is $O(\mathsf{M}(n)\log n)$. Since the maximal multiplicity of an irreducible factor of d is $m-1$, the recursion depth is at most m, and the claim follows.

(iii) With classical arithmetic, the cost for squarefree factorization, partial fraction decomposition, and all Hermite reduction steps is $O(n^2)$ arithmetic operations in F. If $k = \deg d$, then a careful analysis of all computations in (ii) shows that the cost for the first step of Mack's algorithm is $O(n(n-k))$ field operations with classical arithmetic, and summing over all recursive calls yields a total cost of $O(n^2)$ as well.

22.9 The constant coefficient of $\operatorname{res}_x(ay - b', b)$ is $\operatorname{res}_x(-b', b) = \pm\operatorname{res}_x(b, b')$, and this discriminant is nonzero since b is squarefree.

22.11 (iii) Let $h_i = \gcd(b, a - ib')$. The claim follows from the following invariants, which one proves by simultaneous induction on i for $0 \le i \le d$:
(a) $H_i = h_i$ if $i > 0$,
(b) $h_1 \cdots h_i \cdot b_i = b$,
(c) $a \equiv h_1 \cdots h_i (a_i + ib_i') \bmod b_i$.
The case $i = 0$ is immediate. For the induction step, we have

$$\begin{aligned}h_{i+1} &= \gcd(h_1 \cdots h_i b_i, a - (i+1)b') \quad \text{by (b)} \\ &= \gcd(b_i, a - (i+1)b') \quad \text{by (ii)} \\ &= \gcd(b_i, h_1 \cdots h_i(a_i + ib_i') - (i+1)b') \quad \text{by (c)} \\ &= \gcd(b_i, h_1 \cdots h_i(a_i + ib_i') - (i+1)h_1 \cdots h_i b_i') \quad \text{by (b) and Leibniz rule} \\ &= \gcd(b_i, a_i - b_i') = H_{i+1} \quad \text{by (ii)},\end{aligned}$$

proving (a). Claim (b) follows from (a), and (c) follows from

$$\begin{aligned}a_i + ib_i' &= h_{i+1}a_{i+1} + b_i' + ib_i' \\ &\equiv h_{i+1}a_{i+1} + (i+1)h_{i+1}b_{i+1}' \bmod b_{i+1},\end{aligned}$$

again using the Leibniz rule.

22.12 Let $g = \gcd(r,t)$. Then $rU' - sU = t$ and $\gcd(r,s) = 1$ imply that $g \mid U$. Writing $U = gU^*$, we obtain the differential equation $r \cdot (U^*)' - (s - g'r/g) \cdot U^* = t/g$ for U^*. Thus we can set $s^* = s - g'r/g$ and $t^* = t/g$. If $\gcd(r,t^*) > 1$, then we can repeat this process.

22.13 (i) By Lemma 22.18 (iii), we have $\deg H_1 = \deg H_2 = \delta$. If we let $c = \mathrm{lc}(H_1)/\mathrm{lc}(H_2)$, then $H_1 - cH_1 \in S$ and $\deg(H_1 - cH_2) < \delta$, and the lemma implies that $H_1 - cH_2 = 0$.

(ii) If $H^* \in S$ is nonzero, then $H_0 = H^*/\mathrm{lc}(H^*)$ is in S as well, and the claims follow from (i).

(iii) If U_1 is another solution of (8), then $U_1 - U \in S$. Conversely, $U + H$ is a solution of (8), for any $H \in S$.

Chapter 23

23.1 $f(k) = k + \sin(k\pi)$ and $g(k) = 1$.

23.5 (i) For $x \in \mathbb{R}_{\geq 0}$, the function $f(t) = e^{-t}t^{x-1}$ is continuous and strictly positive on the interval $(0,\infty)$, and it is even continuous on $[0,\infty)$ if $x \geq 1$. Let $x \geq 1$. Then there exists a positive $r_x \in \mathbb{R}$ such that $t^{x-1} \leq e^{t/2}$ if $t \geq r_x$. Let $y_x = \int_0^{r_x} f(t)dt \in \mathbb{R}_{\geq 0}$. If $s \geq r_x$, then

$$\begin{aligned} 0 \leq \int_0^s f(t)dt &= \int_0^{r_x} f(t)dt + \int_{r_x}^s f(t)dt \leq y_x + \int_{r_x}^s e^{-t/2}dt \\ &= y_x - 2e^{-s/2} + 2e^{-r_x/2} \leq y_x + 2e^{-r_x/2}. \end{aligned}$$

Thus $\int_0^s f(t)dt$ is bounded for $s \longrightarrow \infty$, and its limit $\int_0^\infty f(t)dt$ is finite.

If $0 < x < 1$, then $\int_1^\infty f(t)dt \leq \int_1^\infty e^{-t}dt = e^{-1}$. For $0 < s < 1$, we have

$$0 \leq \int_s^1 f(t)dt \leq \int_s^1 t^{x-1}dt = \frac{1}{x}(1^x - s^x) \leq \frac{1}{x}.$$

Thus $\int_0^\infty f(t)dt = \lim_{s\longrightarrow 0} \int_s^1 f(t)dt + \int_1^\infty f(t)dt$ is finite as well.

For $x = 0$, the integral does not exist since

$$\int_s^1 f(t)dt \geq e^{-1} \int_s^1 \frac{dt}{t} = -e^{-1} \ln s$$

grows unboundedly for $s \longrightarrow 0$, and the gamma function has a simple pole.

(ii) Using $\frac{\partial}{\partial t}(e^{-t}t^x) = -e^{-t}t^x + xe^{-t}t^{x-1}$ for $x > 0$, we find

$$\begin{aligned} -\Gamma(x+1) + x\Gamma(x) &= -\int_0^\infty e^{-t}t^x dt + x\int_0^\infty e^{-t}t^{x-1}dt \\ &= \int_0^\infty (-e^{-t}t^x + xe^{-t}t^{x-1})dt \\ &= (\lim_{s\longrightarrow\infty} e^{-s}s^x) - (\lim_{s\longrightarrow 0} e^{-s}s^x) = 0. \end{aligned}$$

(iii) We have

$$\Gamma(1) = \int_0^\infty e^{-t}dt = -(\lim_{s\longrightarrow\infty} e^{-s} - e^{-0}) = 1,$$

and the claim follows by induction from (ii).

23.7 (i)

$n \setminus k$	1	2	3	4
1	id			
2	(12)	id		
3	$(123), (132)$	$(12), (13), (23)$	id	
4	$(1234), (1243)$ $(1324), (1342)$ $(1423), (1432)$	$(123), (132), (124)$ $(142), (134), (143)$ $(234), (243), (12)(34)$ $(13)(24), (14)(23)$	$(12), (13)$ $(14), (23)$ $(24), (34)$	id

(ii) The only permutation on n numbers with n cycles is id, and $\left[{n \atop n}\right] = 1$. The permutations with $n-1$ cycles are the transpositions that exchange two numbers $i < j$ and fix all others. There are $\binom{n}{2}$ of them, and hence $\left[{n \atop n-1}\right] = \binom{n}{2}$. Finally, the permutations with exactly one cycle are the cyclic permutations. For such a permutation π, there are $n-1$ choices for $\pi(1)$, namely all numbers except 1, $n-2$ choices for $\pi(\pi(1))$, namely all numbers except 1 and $\pi(1)$, $n-3$ choices for $\pi^3(1)$, and so on, in total $(n-1)!$ choices. Thus $\left[{n \atop 1}\right] = (n-1)!$.

(iii) S_n is the group of all permutations of $\{1,\ldots,n\}$. Consider the map $\varphi: S_n \longrightarrow S_{n-1}$ with $\varphi(\pi) = \sigma$, where $\sigma(i) = \pi(i)$ if $i \notin \{\pi^{-1}(n), \pi(n)\}$, and $\sigma(\pi^{-1}(n)) = \pi(n)$ if $\pi(n) \neq n$. Let $\pi \in S_n$ have k cycles. If $\pi(n) = n$, then $\varphi(\pi)$ is the restriction of π to $\{1,\ldots,n-1\}$ and has $k-1$ cycles. Thus φ maps the elements of S_n with k cycles and n as a fixed point bijectively onto the elements of S_{n-1} with $k-1$ cycles. Now consider those $\pi \in S_n$ that have k cycles and for which $\pi(n) \neq n$. For any $\sigma \in S_{n-1}$ with k cycles, precisely $n-1$ such π are mapped to σ by φ. Thus $\left[{n \atop k}\right] = \left[{n-1 \atop k-1}\right] + (n-1)\left[{n-1 \atop k}\right]$ for $1 \leq k \leq n$.

(iv) We proceed by induction on m. For $m = 0$, we have $x^{\underline{m}} = 1 = (-1)^0\left[{0 \atop 0}\right]$. If $m > 0$, then

$$\begin{aligned} x^{\underline{m}} &= (x-m+1)x^{\underline{m-1}} = (x-m+1)\sum_{0\leq i<m} (-1)^{m-1-i}\left[{m-1 \atop i}\right] x^i \\ &= \sum_{1\leq i\leq m} (-1)^{m-i}\left[{m-1 \atop i-1}\right] x^i + \sum_{0\leq i<m} (-1)^{m-i}(m-1)\left[{m-1 \atop i}\right] x^i \\ &= \sum_{0\leq i\leq m} (-1)^{m-i}\left[{m \atop i}\right] x^i, \end{aligned}$$

by (iii). Using $x^{\overline{m}} = (-1)^m (-x)^{\underline{m}}$, we find that

$$x^{\overline{m}} = \sum_{0 \le i \le m} \begin{bmatrix} m \\ i \end{bmatrix} x^i.$$

(v) This follows from plugging the formula from (iv) into (5) and vice versa.

23.8 In the 1999 edition, the text of this exercise contains some typos, and we first give a corrected version of it.

For $m \in \mathbb{N}$, the mth **Bernoulli number** $B_m \in \mathbb{Q}$ is recursively defined by $B_0 = 1$ and

$$\sum_{0 \le i \le m} \binom{m+1}{i} B_i = 0 \text{ for } m \in \mathbb{N}_{\ge 1},$$

and for $m \ge 0$ we define the polynomial

$$S_m = \frac{1}{m+1} \sum_{1 \le k \le m+1} \binom{m+1}{k} B_{m+1-k} x^k \in \mathbb{Q}[x].$$

(i) Compute B_m and S_m for $0 \le m \le 4$.

(ii) For nonnegative integers $c \le b \le a$, prove the identity

$$\binom{a}{b}\binom{b}{c} = \binom{a}{c}\binom{a-c}{b-c}.$$

(iii) Prove that $\Delta S_m = x^m$ for all $m \in \mathbb{N}$. (Hint: Use (ii).) Show that this implies $\sum_{0 \le k < n} k^m = S_m(n)$ for all $m \in \mathbb{N}$.

(iv) Conclude from Exercise 23.7 and (7) that

$$\frac{B_{m+1-k}}{m+1}\binom{m+1}{k} = \sum_{k-1 \le i \le m+1} \frac{(-1)^{i+1-k}}{i+1} \left\{ m \atop i \right\} \left[i+1 \atop k \right]$$

holds for all $k, m \in \mathbb{N}$ with $1 \le k \le m+1$.

Solution:

(i)

m	B_m	S_m
0	1	x
1	$-\frac{1}{2}$	$\frac{1}{2}x^2 - \frac{1}{2}x$
2	$\frac{1}{6}$	$\frac{1}{3}x^3 - \frac{1}{2}x^2 + \frac{1}{6}x$
3	0	$\frac{1}{4}x^4 - \frac{1}{2}x^3 + \frac{1}{4}x^2$
4	$\frac{1}{30}$	$\frac{1}{5}x^5 - \frac{1}{2}x^4 + \frac{1}{3}x^3 - \frac{1}{30}x$

(ii) We have

$$\begin{aligned}\binom{a}{b}\binom{b}{c} &= \frac{a!}{b!\cdot(a-b)!} \cdot \frac{b!}{c!\cdot(b-c)!} = \frac{a!}{c!\cdot(a-c)!} \cdot \frac{(a-c)!}{(b-c)!\cdot(a-b)!} \\ &= \binom{a}{c}\binom{a-c}{b-c}.\end{aligned}$$

(iii) The recursion formula defining B_{m-1} implies that

$$\sum_{0\le i\le m}\binom{m}{i}B_i = B_m + \sum_{0\le i\le m-1}\binom{m}{i}B_i = B_m \text{ if } m\neq 1,$$

and for $m=1$ the sum is equal to $1+B_m$. Thus

$$\begin{aligned}
(m+1)S_m(x+1) &= \sum_{1\le k\le m+1}\binom{m+1}{k}B_{m+1-k}(x+1)^k \\
&= \sum_{1\le k\le m+1}\sum_{0\le i\le k} B_{m+1-k}\binom{m+1}{k}\binom{k}{i}x^i \\
&= -B_{m+1} + \sum_{0\le i\le m+1}\binom{m+1}{i}x^i \sum_{i\le k\le m+1}\binom{m+1-i}{k-i}B_{m+1-k} \\
&= -B_{m+1} + \sum_{0\le i\le m+1}\binom{m+1}{i}x^i \sum_{0\le k\le m+1-i}\binom{m+1-i}{k}B_k \\
&= -B_{m+1} + \sum_{0\le i\le m+1}\binom{m+1}{i}x^i\cdot B_{m+1-i} + \binom{m+1}{m}x^m \\
&= (m+1)(S_m(x)+x^m).
\end{aligned}$$

(iv) Let $T_m = \sum_{0\le i\le m}\left\{{m \atop i}\right\}x^{\underline{i+1}}/(i+1)$. Then (iii) and (7) imply that $\Delta S_m = x^m = \Delta T_m$, and hence $S_m - T_m$ is a constant, by Lemma 23.3 (vi). Plugging in $x=0$ yields $S_m = T_m$. Using Exercise 23.7 (iv), we find

$$\begin{aligned}
\sum_{1\le k\le m+1}\binom{m+1}{k}\frac{B_{m+1-k}}{m+1}x^k = S_m = T_m &= \sum_{0\le i\le m}\left\{{m \atop i}\right\}\frac{x^{\underline{i+1}}}{i+1} \\
&= \sum_{0\le i\le m}\frac{1}{i+1}\left\{{m \atop i}\right\}\sum_{0\le k\le i+1}(-1)^{i+1-k}\left[{i+1 \atop k}\right]x^k \\
&= \sum_{0\le k\le m+1}x^k\sum_{k-1\le i\le m+1}\frac{(-1)^{i+1-k}}{i+1}\left\{{m \atop i}\right\}\left[{i+1 \atop k}\right],
\end{aligned}$$

and the claim follows by comparing coefficients.

23.9 (i) If we fix the number of women to be $i\in\mathbb{N}$, then there are $\binom{r}{i}$ possibilities to choose the women and $\binom{s}{m-i}$ to choose the men, in total

$$\sum_{0\le i\le m}\binom{r}{i}\binom{s}{m-i}$$

possibilities. On the other hand, there are $\binom{r+s}{m}$ possibilities to choose m persons out of $r+s$ many.

(iii) If we write the binomial coefficient $\binom{n}{k}$ as $n(n-1)\cdots(n-k+1)/k!$ and (formally) substitute indeterminates x and y for r and s, respectively, in the difference of the two sides of (25), we get a polynomial

$$f = \frac{(x+y)^{\underline{m}}}{m!} - \sum_{0\le i\le m} \frac{x^{\underline{i}} y^{\underline{m-i}}}{i!(m-i)!} \in \mathbb{Q}[x,y]$$

of total degree at most m. Vandermonde's convolution can now be restated as

$$f(r,s) = 0 \text{ for all } r,s \in \mathbb{N}.$$

Lemma 6.44, applied to f and $S = \{1,\ldots,m+1\}$, then implies that f is the zero polynomial.

(iv) The binomial theorem follows from (iii) by multiplying with $m!$. The rising factorials also satisfy a binomial theorem:

$$\sum_{0\le i\le m} \binom{m}{i} x^{\overline{i}} y^{\overline{m-i}} = (x+y)^{\overline{m}}.$$

This follows from the binomial theorem for the falling factorials by using $x^{\overline{i}} = (-1)^i(-x)^{\underline{i}}$.

23.10 (i) We have

$$\begin{aligned} x^{\underline{m+n}} &= x(x-1)\cdots(x-m+1)(x-m)(x-m-1)\cdots(x-m-n+1) \\ &= x^{\underline{m}}(E^{-m}x)(E^{-m}(x-1))\cdots(E^{-m}(x-n+1)) = x^{\underline{m}}E^{-m}x^{\underline{n}}. \end{aligned}$$

(ii) The definition reads

$$x^{\overline{-n}} = \frac{1}{(x+n)^{\underline{n}}} = \frac{1}{(x+1)^{\overline{n}}} = \frac{1}{(x+1)(x+2)\cdots(x+n)} \tag{30}$$

for all $n \in \mathbb{N}$.

To prove (26) for arbitrary integers m,n, we distinguish several cases. The case $m,n \ge 0$ has been shown in (i). For example, if $n < 0$ and $m+n \ge 0$, then $m > 0$, and hence

$$\begin{aligned} x^{\underline{m+n}} &= x(x-1)\cdots(x-m-n+1) \\ &= \frac{x(x-1)\cdots(x-m-n+1)(x-m-n)(x-m-n-1)\cdots(x-m+1)}{(x-m-n)(x-m-n-1)\cdots(x-m+1)} \\ &= \frac{x^{\underline{m}}}{(x-m+1)^{\overline{-n}}} = x^{\underline{m}}(x-m)^{\underline{n}}, \end{aligned}$$

by (30). The proof for the other cases is similar.

If $m < 0$, then

$$\begin{aligned}\Delta x^{\underline{m}} &= \frac{1}{(x+2)\cdots(x-m)(x-m+1)} - \frac{1}{(x+1)(x+2)\cdots(x-m)} \\ &= \frac{(x+1)-(x-m+1)}{(x+1)(x+2)\cdots(x-m)(x-m+1)} = m x^{\underline{m-1}}.\end{aligned}$$

23.11 (i) $(n^4 - 6n^3 + 11n^2 + 14n)/4$; (ii) $n2^n - 2^{n+1} + 2$.

23.12 For a polynomial $f \in F[x]$, we have $\deg(\Delta f) \le \deg f - 1$, with equality if $f \notin F$. Now

$$\Delta\left(\frac{f}{g}\right) = \frac{Ef}{Eg} - \frac{f}{g} = \frac{Ef\cdot g - f\cdot Eg}{g\cdot Eg} = \frac{\Delta f\cdot g - f\cdot \Delta g}{g\cdot Eg}.$$

Now the degree of the numerator of that expression is at most $\deg f + \deg g - 1$, and the coefficient of $x^{\deg f+\deg g-1}$ in it is $\operatorname{lc}(f)\operatorname{lc}(g)(\deg f - \deg g)$. Thus the degree of the numerator is equal to $\deg f + \deg g - 1$ if and only if $\deg f = \deg g$, or equivalently, $\deg \rho = 0$, and the claim follows since the denominator has degree $2\deg g$. In particular, we have $\deg(\Delta\rho) = \deg\rho - 1 \ne -1$ if $\deg\rho \ne 0$, and $\deg(\Delta\rho) < \deg\rho - 1 < -1$ if $\deg\rho = 0$.

23.13 (i) We have $D(\Delta\rho) = D(\rho(x+1) - \rho(x)) = (D\rho)(x+1) - (D\rho)(x) = \Delta(D\rho)$, by the chain rule for the differential operator.

(ii) Assuming (i), we have

$$\begin{aligned}\Delta\Psi_m(x) &= \Delta D^m \ln\Gamma(x) = D^m \Delta \ln\Gamma(x) = D^m \ln\left(\frac{\Gamma(x+1)}{\Gamma(x)}\right) \\ &= D^m \ln x = D^{m-1}x^{-1} = (-1)^{m-1}(m-1)!\,x^{-m}.\end{aligned}$$

(iii) The partial fraction decomposition is $1/(x^2+ax) = (1/x - 1/(x+a))/a$, and hence

$$\begin{aligned}\Sigma\frac{1}{x^2+ax} &= \frac{1}{a}\left(\Sigma\frac{1}{x} - \Sigma\frac{1}{x+a}\right) = \frac{1}{a}(\Psi_1(x) - \Psi_1(x+a)) \\ &= \frac{1}{a}D\ln\left(\frac{\Gamma(x)}{\Gamma(x+a)}\right) = -\frac{1}{a}D\ln x^{\overline{a}} = -\frac{1}{a}\frac{Dx^{\overline{a}}}{x^{\overline{a}}}.\end{aligned}$$

(In the 1999 edition, this a was called d.)

23.15 (i) $(x(x+3),(x+3)^2,1,x+3)$; (ii) $(x,x+1,1,1,1,x+3)$.

23.16 We only have to verify property (F_4) for $(f_1,\ldots,f_m)$. So we let $1 \le i \le j \le m$. If $i \ge 2$, then property (F_4) for $(g_1,\ldots,g_{m-1})$ implies that

$$\gcd(f_i^{\underline{i-1}}, E^{-j+1}f_j) = \gcd(g_{i-1}^{\underline{i-1}}, E^{-j+1}g_{j-1}) = 1.$$

Now $E^{-j+1}f_j = E^{-j+1}g_{j-1}$ divides f/g, by (14), and Ef_i divides $E(f_1\cdots f_m) = (Ef)/g$, and since f/g and $(Ef)/g$ are coprime, so are $E^{-j+1}f_j$ and Ef_i. Thus $\gcd(Ef_i^i, E^{-j+1}f_j) = 1$, and applying E^{-1} once shows that (F_4) holds if $i \geq 2$. For $i = 1$, we find that $E^{-j+1}f_j \mid f/g$ and $Ef_1 \mid (Ef)/g$, again $\gcd(f/g, (Ef)/g) = 1$ implies that $\gcd(Ef_1, E^{-j+1}f_j) = 1$, and hence also $\gcd(f_1, E^{-j}f_j) = 1$.

23.18 The term ratios are

(i) $\dfrac{(x^2+4x+4)(x^2+4x+3)(x^2+4x+2)}{(x^2+2x+1)(x^2+2x)(x^2+2x-1)}$; (ii) $\dfrac{x+2}{x+1}\cdot 2^{2x+1}$;

(iii) $(-1)^{2x+1}(x+1) = -x-1$.

Thus only (ii) is not hypergeometric; (i) is a polynomial.

23.20 Let $g(k) = k^{\underline{2}} = k(k-1)$. We know from Section 23.1 that

$$\Sigma g(k) = \frac{k^{\underline{3}}}{3} = \frac{(k-2)}{3}g(k). \tag{31}$$

The term ratio is

$$\sigma(k) = \frac{g(k+1)}{g(k)} = \frac{k+1}{k-1}.$$

We take $a = x+1$ and $b = x-1$. Step 1 of Algorithm 23.18 computes

$$R = \operatorname{res}_x(x+1, x+y-1) = \det\begin{pmatrix} 1 & 1 \\ 1 & y-1 \end{pmatrix} = y-2,$$

and thus $d = 2$. In step 2, we have

$$H_1 = \gcd(E^{-1}a, b) = \gcd(x, x-1) = 1,$$
$$H_2 = \gcd(E^{-1}a, Eb) = \gcd(x, x) = x,$$

and hence $V = H_1^1 H_2^2 = x^{\underline{2}} = x(x-1)$. (Algorithm 23.20 produces the same values.) Now (20) is

$$(x+1)x^{\underline{2}}\cdot EU - (x-1)(x+1)^{\underline{2}}U = (x-1)x^{\underline{2}}(x+1)^{\underline{2}},$$

or equivalently

$$EU - U = x(x-1), \tag{32}$$

after dividing both sides by $(x+1)x(x-1)$, which of course is nothing else than our original problem, but we have now found that the denominator of τ (from (15)) divides $V = x(x-1)$.

The following derivation refers to the 2003 edition only. For the determination of $\deg U$, we have $r = s = 1$, $t = x^2 - x$, $\deg r - 1 = -1 > -\infty = \deg(s-r)$, and $\delta = 0$. Lemma 23.24 (i) implies that either $\deg U = \deg t - m = 3$ or $\deg U = \delta = 0$.

The latter is impossible, and we make the ansatz $U = U_3x^3 + U_2x^2 + U_1x + U_0$ in (32) and obtain

$$\begin{aligned} x^2 - x &= (U_3(x^3 + 3x^2 + 3x + 1) + U_2(x^2 + 2x + 1) + U_1(x+1) + U_0) \\ &\quad - (U_3x^3 + U_2x^2 + U_1x + U_0) \\ &= 3U_3x^2 + (3U_3 + 2U_2)x + (U_3 + U_2 + U_1)x, \end{aligned}$$

which leads—by comparing coefficients on both sides—to the system of linear equations

$$1 = 3U_3, \quad -1 = 3U_3 + 2U_2, \quad 0 = U_3 + U_2 + U_1.$$

The solutions are $U_3 = 1/3$, $U_2 = -1$, and $U_1 = 2/3$, with arbitrary $U_0 \in F$. Setting $U_0 = 0$, we have $U = (x^3 - 3x^2 + 2x)/3 = x(x-1)(x-2)/3$, and finally $\tau = U/V = (x-2)/3$, in accordance with (31).

23.21 This solution refers to the 2003 edition. The term ratio of the binomial coefficient is $\sigma(x) = (-x+n)/(x+1)$, and we have $a = -x+n$ and $b = x+1$. As in the Example 23.27, Algorithms 23.20 and 23.18 yield $V = 1$, since the resultant R only changes sign. Equation (20) is

$$(-x+n)EU - (x+1)U = x+1.$$

Lemma 23.24 implies that $\deg U = \deg t - m = 0$. Letting $U = U_0 \in F$, we obtain

$$x+1 = (-x+n)U_0 - (x+1)U_0 = (-2x+n-1)U_0,$$

which has no solution $U_0 \in F$ since $n \neq -1$. Thus $\Sigma\binom{n}{x}$ is not hypergeometric.

23.22 Only the first sum is hypergeometric, and we have

$$\Sigma\left(\frac{3x+1}{x+1}\binom{2x}{x}\right) = \binom{2x}{x}.$$

23.23 This solution refers to the 2003 edition. Neither of the two sums is hypergeometric. The term ratio for $g(x) = (-1)^x\binom{n}{x}^2$ is $\sigma(x) = -(n-x)^2/(x+1)^2$, and hence $a = -x^2 + 2nx - n^2$ and $b = x^2 - 2x + 1$. The resultant

$$\operatorname{res}_x(a(x), b(x+y)) = (y+n+1)^4$$

has no nonnegative integer roots, and both Algorithms 23.20 and 23.18 return $V = 1$. Equation (20) is

$$(-x^2 + 2nx - n^2)EU - (x^2 + 2x + 1)U = x^2 + 2x + 1.$$

Lemma 23.24 implies that $\deg U = \deg t - m = 0$. Comparing leading coefficients gives $U = -1/2$, and comparing coefficients of x yields $U = \frac{1}{n-1}$. This is a contradiction since $n \geq 1$.

If $g(x) = \binom{n}{x}^2$, then we again find $V = 1$, since the resultant does not change, but now (20) is

$$(x^2 - 2nx + n^2)EU - (x^2 + 2x + 1)U = x^2 + 2x + 1, \tag{33}$$

and $\delta = 2n + 2 \in \mathbb{N}$. Thus either $\deg U = \deg t - m = 1$ or $\deg U = \delta = 2n + 2$, by Lemma 23.24 (i).

Assume first that $\deg U = 2n + 2$. The unique nonzero monic solution $U^* \in F[x]$ of the corresponding homogeneous equation

$$(x - n)^2 EU^* - (x + 1)^2 U^* = 0$$

is $U^* = (x^{\underline{n+1}})^2$. Now $U - \mathrm{lc}(U)U^*$ is also a solution of the inhomogeneous equation (33) and has degree less than $2n + 2$, and Lemma 23.24 implies that it has degree 1. Thus it is sufficient to look for a solution of degree 1. This yields

$$(x - n)^2 (U_1(x + 1) + U_0) - (x + 1)^2 (U_1 x + U_0) = (x + 1)^2,$$

or equivalently, the linear system

$$-(2n + 1)U_1 = 2, \quad (n^2 - 2n - 1)U_1 - 2(n + 1)U_0 = 2, \quad n^2 U_1 + (n^2 - 1)U_0 = 1.$$

The first equation gives $U_1 = -n - 1/2$, from the second equation we obtain $U_0 = -\frac{1}{2}(n + 1)/(2n + 1)$, and the third equation yields $U_0 = (n + 1)/(n - 1)(2n + 1)$. The latter two are equal if and only if $n = -1$, which is not the case, and hence (33) has no solution.

23.24 (ii) The only root $-(2n - 1)/2$ of $R = \mathrm{res}_x(a(x), b(x + y))$ is not integral.

(iii) The following derivation refers to the 2003 edition only. Equation (21) is

$$\begin{aligned}(x + 1)^2 \cdot EU - \left(x^2 + (2n + 1)x + \frac{(2n + 1)^2}{4}\right) U \\ = \left(x^2 + (2n + 1)x + \frac{(2n + 1)^2}{4}\right),\end{aligned} \tag{34}$$

and we have $r = a$ and $s = t = b$. Moreover,

$$\deg r - 1 = 1 = \deg((2n - 1)x + (4n^2 + 4n - 3)/4) = \deg(s - r),$$

$\delta = 2n - 1 \in \mathbb{N}$, and Lemma 23.24 (i) says that either $\deg U = \deg t - m = -1$, which is impossible, or $\deg U = \delta = 2n - 1 \geq 0$. (Thus the summation problem has no solution if n is an indeterminate.) The value of δ is exponentially large in the size of a and b, which is about $\log_{2^{64}} n$ words.

(iv) In terms of the operator L, (34) reads $LU = b$. If $Lf = 0$ for some nonzero $f \in \mathbb{Q}[x]$, then $a/b = f/Ef$, which is a contradiction since R has no integral roots, and hence L is injective. By construction, we have $\deg(Lf) \le 1 + \deg f$, and even $\deg(Lf) \le \deg f$ if $\deg f = 2n-1$. Thus L maps the $2n$-dimensional vector space $W \subseteq \mathbb{Q}[x]$ of all polynomials of degree less than $2n$ to itself, and since L is injective and W is finite-dimensional, L is also surjective on W. Finally, we have $\deg b = 2 < 2n$, whence $b \in W$, and there is a unique polynomial $U \in W$ of degree $2n-1$ such that $LU = b$.

(v) For $n = 6$, we have $\sum_{0 \le k < m} g(k) = U(m)g(m) - U(0)g(0)$ for all $m \in \mathbb{N}$, where

$$\begin{aligned} U = \frac{(4x^2+44x+121)(4x+7)}{281\,302\,875}(&8\,388\,608x^8 + 117\,440\,512x^7 + 658\,767\,872x^6 \\ &+1\,881\,800\,704x^5 + 2\,862\,755\,840x^4 + 2\,179\,846\,144x^3 + 648\,167\,040x^2 \\ &+504\,000x - 496\,125). \end{aligned}$$

23.26 Let $g = \gcd(r,t)$. Then $r \cdot EU - s \cdot U = t$ and $\gcd(r,s) = 1$ imply that $g \mid U$. Writing $U = gU^*$, we obtain the difference equation $(r(Eg)/g) \cdot EU^* - s \cdot U^* = t/g$ for U^*. Similarly, if $g = \gcd(s,t) > 1$, then $g \mid EU$, and writing $U = E^{-1}gU^*$, we obtain the difference equation $r \cdot EU^* - (s(E^{-1}g)/g) \cdot U^* = t/g$ for U^*.

23.27 By Exercise 6.23, the roots of f and g in $\mathbb{C}$ are absolutely at most $2B$. By the discussion preceding Example 23.22 on page 653 (in the 2003 edition), we conclude that

$$d \le \max\{|\beta - \alpha| : f(\alpha) = 0 = g(\beta)\} \le 4B.$$

Using Mignotte's bound (Corollary 6.33) would lead to the slightly worse estimate $4(\max\{\deg f, \deg g\} + 1)^{1/2}B$.

23.28 In the 1999 edition, there is a typo in the exercise; we have to require that $\deg g > 0$ instead of $\deg f > 0$.

(i) We write $\rho = f/g$, with $f, g \in F[x]$ coprime and $\deg g > 0$. Then

$$\Delta\rho = \frac{g \cdot Ef - f \cdot Eg}{g \cdot Eg} = \frac{u}{v},$$

where $u = (g \cdot Ef - f \cdot Eg)/\gcd(g, Eg)$ and $v = (g \cdot Eg)/\gcd(g, Eg) = \operatorname{lcm}(g, Eg)$ are coprime. Let $d = \operatorname{dis}(\rho)$ and $p \in F[x]$ be an irreducible factor of $\gcd(g, E^d g)$. Then $Ep \mid Eg \mid v$ and $Ep \mid E^{d+1}g \mid E^{d+1}v$, which shows that $\operatorname{dis}(\Delta\rho) \ge d+1$. On the other hand, if p is an irreducible factor of $\gcd(v, E^k v)$ for some $k \in \mathbb{N}$, then $p \mid v \mid g \cdot Eg$ and $p \mid E^k v \mid E^k g \cdot E^{k+1} g$. Thus $\gcd(g, E^l g) \ne 1$ for some $l \in \{k-1, k, k+1\}$. In particular, $k - 1 \le l \le d$, which implies that $\operatorname{dis}(\Delta\rho) \le d+1$.

(ii) Since the difference of a polynomial is again a polynomial, ρ is not a polynomial. But then (i) implies that $0 = \operatorname{dis}(x^{-m}) = \operatorname{dis}(\rho) - 1 \le -1$, and this contradiction shows that no such ρ exists.

23.31 We rewrite (21) as

$$r \cdot \Delta U - (s-r) \cdot U = t$$

and compare the degrees and the top coefficients. Firstly, we have

$$\begin{aligned} \deg t &\le \max\{\deg(rU'), \deg((s-r)U)\} \\ &\le \max\{\deg r + \deg U - 1, \deg(s-r) + \deg U\} = m + \deg U. \end{aligned}$$

Let $\gamma \in F$ denote the coefficient of x^{m+1} in r. Then the coefficient of $x^{m+\deg U}$ in rU' is $\gamma \operatorname{lc}(U) \deg U$, and the coefficient of $x^{m+\deg U}$ in $(s-r)U$ is $\delta \operatorname{lc}(U)$. Thus the coefficient of $x^{m+\deg U}$ in t is $(\gamma \deg U - \delta) \operatorname{lc}(U)$, and $\deg t < m + \deg U$ if and only if this coefficient vanishes.

If $\deg r - 1 < \deg(s-r)$, then $\gamma = 0$ and $\delta = \operatorname{lc}(s-r) \neq 0$, and hence $\deg U = \deg t - m$. Otherwise, we have $\gamma = \operatorname{lc}(r) = 1$. We conclude that $\deg U \ge \deg t - m$, with strict inequality if and only if $\deg r - 1 \ge \deg(s-r)$ and $\deg U = \delta$. This proves (i), (ii), and (iii).

To show (iv), we assume that $U^* \in F[x]$ is another solution of (21). Then the homogeneous equation $r(U-U^*)' - (s-r)(U-U^*) = 0$ holds for the difference $U - U^*$, and the claim follows from (iii).

Chapter 24

24.3 We first note that $kp_k = 0$ if $k < s$ or $k > n-w+s$, and hence the summation range on the left hand side in (6) may be replaced by $s \le k \le n-w+s$. Multiplying both sides by $\binom{n}{w}/s$, we find that (6) is equivalent to

$$\sum_{s \le k \le n-w+s} \binom{k}{s} \binom{n-k}{w-s} = \frac{n+1}{w+1} \binom{n}{w} = \binom{n+1}{w+1}. \tag{30}$$

If $n = w$, then both sides of (30) are equal to 1. Now let $n > w = s$, and assume that the claim has already been shown for $n-1$. Then

$$\begin{aligned} \sum_{s \le k \le n} \binom{k}{s} \binom{n-k}{s-s} &= \binom{n}{s} + \sum_{s \le k \le n-1} \binom{k}{s} \binom{n-1-k}{s-s} \\ &= \binom{n}{s} + \binom{n}{s+1} = \binom{n+1}{s+1}. \end{aligned}$$

Thus (30) is true if $n = w$ or $w = s$. Now assume that $w > s$, and that the claim has already been shown for $w-1$ and arbitrary $n \ge w-1$. We have already seen that it is true for $n = w$, and hence we may also assume that $n > w$ and that the claim

holds for $n-1$ and w. Then

$$\begin{aligned}
&\sum_{s\le k\le n-w+s}\binom{k}{s}\binom{n-k}{w-s}\\
&\qquad=\binom{n-w+s}{s}+\sum_{s\le k\le n-w+s-1}\binom{k}{s}\left(\binom{n-k-1}{w-s}+\binom{n-k-1}{w-s-1}\right)\\
&\qquad=\sum_{s\le k\le n-1-w+s}\binom{k}{s}\binom{n-1-k}{w-s}+\sum_{s\le k\le n-1-(w-1)+s}\binom{k}{s}\binom{n-1-k}{w-1-s}\\
&\qquad=\binom{n}{w+1}+\binom{n}{w}=\binom{n+1}{w+1}.
\end{aligned}$$

24.4 (ii) Let $I\subseteq\{1,\ldots,n\}$ be nonempty. If $\sum_{i\in I}\lambda_i a_i=0$, with all $\lambda_i\in F$, then the linearity of the inner product in the first argument implies that $\sum_{i\in I}\lambda_i(a_i\star a_j)=0$ for $1\le j\le n$. Conversely, let $\sum_{i\in I}\lambda_i(a_i\star a_j)=v\star a_j=0$ for $1\le j\le n$, where $v=\sum_{i\in I}\lambda_i a_i$. Then also $v\star v=\sum_{j\in I}\lambda_j(v\star a_j)=0$, and hence $v=0$. Thus the a_i for $i\in I$ are linearly independent if and only if the rows of G with index in I are.

24.5 (i) The projection of A is $\{(u,v)\in\mathbb{R}^2\colon\exists w\in\mathbb{R}\ (u,v,w)\in A\}$. This set is contained in B. For the reverse inclusion, we let $u,v\in[-1,1]$ such that $u+v\le -2/3$. Then $u,v\le 1/3$, and if we let $w=-1-(u+v)$, we find $-1/3\le w\le 1$, $u+w=-1-v\le -2/3$, and $v+w=-1-u\le -2/3$. Thus $(u,v,w)\in A$.

(ii) The roots of the polynomial $9u^2+6u-23$ are $(-1+2\sqrt{6})/3\ge 1.29965$ and $(-1-2\sqrt{6})/3\le -1.96633$, and hence $9u^2+6u-23=0$ comprises two parallel vertical lines enclosing, but not intersecting, B.