9. Exercise sheet
Hand in solutions until Thursday, 14 June 2007.

Exercise 9.1 (Hidden message). (8 points)

Once again a new mission is waiting for her Majesty’s finest agent. Old Q has received an assignment from M to find a way how 007 may send a secret message to the London headquarters unnoticed.

In the guise of a broker James Bond has easy access to the Internet. Q has learned that, at the stock market, buyers’ and sellers’ orders are signed using the ElGamal signature scheme. The mastermind of the Q-Branch starts from there:

Q: Here is the solution, 007. Naturally you are well acquainted with the signing of electronic messages using the ElGamal scheme.

007: I have read the Russian translation of the article, Q.

Q: Splendid! We will use this scheme to hide the message you want to send to M. The present system uses the prime number \( p = 311303 \) and the group \( \mathbb{Z}_p^* \). The element \( g = 5 \) is the generator of \( \mathbb{Z}_p^* \) that was adopted. The secret part of the key is \( \alpha = 34567 \). Is everything quite clear so far, 007?

007: Yes, Q. Everything quite standard. So where is the trick?

Q: 007, for the first time you are showing some interest in my work! Instead of the random number \( \beta \) used for signing the message \( m \) you will use your secret message \( \hat{m} \). This is the date (formatted TTMMJJ) on which we — how would you put this — must be prepared for a surprise. Good luck, 007!

(i) What is/are the “conventional” purpose(s) of a randomly chosen component for a digital signature (e.g. the \( \beta \) in the ElGamal scheme)?

(ii) Explain why Q assumes that the transmission of \( \hat{m} \) is secure.

(iii) After some time Q receives the following signature: \( (54321, 6193, 132622) \). Check whether this message originates with 007. What is the date that 007 predicts for the surprise?

(iv) Which conditions (with respect to the variables) must be met so that this computation works?
Exercise 9.2 (Attacks on the ElGamal signature scheme).  (4 points)

After prior failures princess Jasmin and Genie have been doing a lot of thinking and research. Genie has proposed to use the ElGamal signature scheme. They have chosen the prime number $p = 1,334,537$ and the generator $g = 3$. The public key of the princess Jasmin is $a = 143,401$.

(i) They have sent the message $(x, b, \gamma) = (7,654,335,037,820,465)$. Unfortunately, Genie was not very careful. He wrote down the number $\beta$ somewhere and forgot to burn the piece of paper after calculating the signature. Now Jaffar knows the number $\beta = 377$. Compute the secret key $\alpha$.

(ii) Princess Jasmin has changed her secret key. She now has the public key $a = 568,267$. This time Jaffar could not find the number $\beta$. Because of this he used an enchantment so that Jasmin’s random number generator has output the same value for $\beta$ twice in a row. This was the case for the messages $(2,001,576,885,1,323,376)$ and $(234,576,885,1,161,723)$. Now compute Jasmin’s secret key $\alpha$.

Exercise 9.3 (Expected runtime).  (8+4 points)

Algorithm. Loop.

Input: None.
Output: The runtime $N$.

1. $N \leftarrow 0$,
2. Repeat
3. $N \leftarrow N + 1$,
4. Until $\text{rnd}() = 0$
5. Return $N$

Algorithm. TWO.

Input: Some parameter $k \in K$.
Output: The runtime $N_k$.

1. $N \leftarrow 0$,
2. Repeat 3–4
3. $N \leftarrow N + 1$,
4. $m \leftarrow \text{rnd}()$
5. Until $h(m) = k$
6. Return $N$

Consider the algorithm Loop where the probability that $\text{rnd}() = 0$ is exactly $p$ in each round. Denote $q := \text{prob}(\text{rnd}() \neq 0) = 1 - p$.

(i) Compute $\text{prob}(N = n)$. (You might want to consider $\text{prob}(N = 1)$, $\text{prob}(N = 2)$, $\text{prob}(N = 3)$, first.)

(ii) Show that the expected value of $N$, ie. $E(N) = \sum_{n \in \mathbb{N}} n \text{prob}(N = n)$, equals $\frac{1}{p}$.

Recall: $\sum nq^{n-1}$ is the derivative of the limit of the geometric series $\sum q^n$ with respect to $q$, and the latter is $\frac{1}{1-q} = \frac{1}{p}$. 
What happens if the probabilities are not always the same? In the course we have considered the case of guessing a second preimage for a hash function $h: \{0,1\}^* \rightarrow K$. Two where the probability $p_k$ for $h(rnd()) = k$ may depend on $k$. Actually, we consider the case where $p_k$ is the same as the probability that $k$ occurs as an input; in particular, $\sum_{k \in K} p_k = 1$. As above, we obtain $E(N_k) = \frac{1}{p_k}$.

(iii) Assume that any input is chosen with the same probability $p_k = 1/\#K$. What is the average runtime?

(iv) Consider $K = \{1, 2\}$ and $p_1 = p \in ]0,1[$, $p_2 = q = 1 - p$. What is the average runtime now? Evaluate it for the unbalanced value $p = 1/501$ and compare to (iii).

(v) Assume that each $K$ occurs almost uniformly in the sense that $p_k \geq \frac{1}{2\#K}$. Tightly bound the runtime now.

(vi) Relax the condition ‘almost uniform’ with a still reasonable (What’s reasonable?) bound on the runtime.

Exercise 9.4 (Hash crisis). (0 points)

Study SHA-1, the recent attacks, and devise a new fast hash function invulnerable to the known attacks.