B-it **Bridging Course (SS05)**

Lecturer: L. El Aimani

Assignments

1 Assignment 1

1.1 Find-Union algorithms

Implement the Find-Union algorithms studied in class.

1.2 Solving Recurrences

solve the following recurrences:

1. $C_N = C_{\frac{N}{2}} + N$ for $N \geq 2$ with $C_1 = 0$.
2. $C_N = 2C_{\frac{N}{2}} + N$ for $N \geq 2$ with $C_1 = 0$.
3. $C_N = 2C_{\frac{N}{2}} + 1$ for $N \geq 2$ with $C_1 = 1$.

1.3 Some recursive algorithms

1. Write the factorial function (iterative and recursive version) and evaluate its cost.
2. The gcd of two integers is defined to be the largest common divisor of the given integers. An algorithmic solution for computing the gcd of two integers a and b is based on the following remark: $\gcd(a, b) = \gcd(b, a \bmod b)$,where $a \geq b$. Write the recursive and iterative version of the Euclidean algorithm, which is based on the above remark.
3. Given a sorted array of integers, write the sequential and binary search function of an item in the array.
4. Write a recursive program to evaluate prefix expressions.

1.4 The Abstract Data Structure: List

We define the abstract data type **RList** as a collection of Elements (predefined type) that has the following operations:

- boolean IsEmpty(RList l): returns true if the RList l is empty and false otherwise.
- Element Head(RList l): returns the first Element of the RList. This operation is not defined for empty RLists.
- RList End (RList l): returns the RList after removing the first Element. Again, the argument l should not be the empty RList.

Write the functions Member that checks whether a given Element in a given RList and the function IsSet that tests whether a given RList is a set of Elements.

2 Assignment 2

2.1 Abstract Data type: Point

- Write the implementation of the data type Point.
- Write a client program that computes the closest points regarding a certain distance.

2.2 Recursive lists

1. **Concatenation of lists**
   We define concat: list × list → list as the concatenation of two lists. Complete the following axioms:
   - length(concat(l, m))
   - i^{th}(concat(l, m), j)
   - concat(empty_list, l)
   - concat(cons(e, l), m)
2. **Search of an element (present) in a list**
   We define search: list × element → position. Complete the following axioms:
   - content(search(l, e))
   - search(cons(e,l), e)
   - e ≠ f, search(cons(e, l), f)
3. **Implementation**
   Give a linked list implementation of a recursive list.

2.3 Pushdown stack

- Give a linked list implementation of a pushdown stack.
- Write a program that reads any postfix expression involving multiplication and addition of integers, then evaluates the expression and prints the computed result.

2.4 Mathematical properties of binary trees

1. Prove by induction that the external path length of any binary tree with N internal nodes is 2N greater than the internal path length.
2. Prove that the internal path length of a binary tree with N internal nodes is at least Nlog(N/4) and at most N(N-1)/2.

3 Assignment 3

3.1 Tree Traversal

Perform the preorder, inorder, postorder traversals of the tree studied in class.

3.2 Implementation

Implement the postorder traversal of a binary tree.

4 Assignment 4

4.1 Sorting using a BST

- construct a BST from the following set {a,s,o,r,t,i,n,g,e,x,a,m,p,l,e} by using successive insertions at the leaf level (as seen in class).
- sort the above set by using an appropriate tree traversal.
- deduce a method for sorting using a BST and analyse its cost.

4.2 Remove operation in a BST

Write the remove operation in a BST. You will need for that an operation max that returns the maximum element in a BST, and a $m\bar{a}x$ function that returns a BST from which we removed its maximum element.

4.3 Quicksort

Implement the Quicksort.