ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Algorithm

8 November 2010

- The only authorized document is a $A4$ sheet duplex.
- Calculators, telephones, computers etc ... are prohibited.

Nom :

Prénom :

Section :

Exercice 1 / 10 points	Exercice 2 / 15 points	Exercice 3 / 25 points	Exercice 4 / 15 points

Total / 65

Problem 1 [10 points].

Let $A(n) = \sum_{k=1}^{n} k \cdot k!$.

1. Calculate $A(n)$ for $n = 1, 2, 3, 4, 5$.
2. Compare the sequence of $A(n)$, $n = 1, 2, 3, 4, 5$ with the sequence $n!$ for $n = 1, 2, 3, 4, 5$.
3. Guess a general formula for $A(n)$ and prove your formula via induction.

Problem 2 [15 points].

Given $S \subseteq \mathbb{N}$ and $x \in \mathbb{N}$ consider this problem: is there $T \subseteq S$, so that $x = \sum_{i \in T} i$? (This problem is called the *subset sum* problem)

1. Provide a formal specification of this problem.
2. Assume S is given as $\{s_1, s_2, \ldots, s_n\}$ with the property that $s_2 > s_1$, $s_3 > s_2 + s_1$, in general $s_k > \sum_{i=1}^{k-1} s_i$ for all k. Give an algorithm which finds the answer in at most n additions. Explain why it works.

Problem 3 [25 points].

1. Suppose we are only allowed to use the stack data structure. Is it possible to implement a queue by using two stacks? If yes, give the algorithms for the corresponding queue operations.

2. Again, we want to implement a queue by using two stacks, but with the restriction that the running time of the operations should be similar to that of the usual implementation of the queue.

   Precisely, suppose that the stack operations are performed in $O(1)$ time. (i.e. it does not depend on the size of the stack) Write down the algorithms for the operations of the queue based on two stacks which have the following property: If we start with an empty queue then we need $O(m)$ stack operations to perform the first m queue operations.

Problem 4 [15 points].

Let T be an AVL binary search tree of height h (recall that bal(k) is defined as the height of the left sub-tree minus the height of the right sub-tree of the vertex k; in an AVL search tree, for each vertex k, bal(k) $\in \{0, +1, -1\}$). An element is inserted to T, and a binary search tree T' is obtained. Answer the following questions concerning T' *before any rotations are performed*.

1. What is the *smallest* possible number of vertices with bal(k) $\in \{+2, -2\}$? Briefly explain your answer.

2. What is the *largest* possible number of vertices with bal(k) $\in \{+2, -2\}$? Show an example with this number and explain why it is the largest.

3. Let s be a vertex in T', and t be its left-hand child. Is it possible that bal(t) $= +2$ and bal(s) $= -1$? If yes, show an example. If not, explain why.