Aalto University School of Science
Department of Computer Science and Engineering
T-106.1223/CSE-A1141 Data Structures and Algorithms Y

EXAMINATION
8.12.2014

Write on each paper your name, student number, degree programme, and the course code with name. Also write the date, hall, the number of papers you return, and your *signature*. Using any extra devices is prohibited in this examination.

1) Ten Questions (10 x 1p + 1p = 11p)

This is a *compulsory part* of the final exam. You need to get at least 5p out of the maximum 10p so that the rest of the exam will be checked. However, this part alone is not enough to pass the whole exam. On the other hand, in order to get 5p, you are not required to give "the exactly correct answer", but more or less show that *you have understood the functionality of the code fragments* related to this part. Thus, pay attention to the reasoning. Refer to the code line numbers if possible.

In the following, you can see two algorithms computing the factorial. Read through all the questions below without answering them and after that familiarize yourself with the code throughout. After this, answer all the questions and take time for pondering and explaining your reasoning. Note, however, that all the questions refer to the given algorithms. In addition, the claims in the questions can be justified to be either true or false, thus the *argumentation* is the only thing that matters for the points!

```
1 int fact_1(int n)
2 {
3     if (n<2)
4         return 1;
5     else
6         return fact_1(n-1)*n;
7 }
```

```
 8 int fact_2(int n)
 9 {
10     int i,fact;
11     fact = 1;
12     for (i=1; i<=n; i++)
13         fact = fact*i;
14     return fact;
15 }
```

a) *Describe* how Algorithm 1 (i.e., fact_1) works by using an appropriate *example*.
b) *Describe* how Algorithm 2 (i.e., fact_2) works by using an appropriate *example*.
c) *In which order and how many* multiplications fact_1 does? *Give an example* in case the algorithm is called with parameter n=3.
d) *In which order and how many* multiplications fact_2 does? *Give an example* in case the algorithm is called with parameter n=3.
e) *Analyse* the time complexity of Algorithm 1 in terms of the input size n.
f) *Analyse* the time complexity of Algorithm 2 in terms of the input size n.
g) *Argue* whether it is true or false: Algorithm 1 is more efficient than Algorithm 2.
h) *Argue* whether it is true or false: Algorithm 1 computes the same function than Algorithm 2.
i) *What is the order of multiplications* in Algorithm 1 if the line 6 would be changed to "return n*fact_1(n-1);"? *Give an example*.
j) Is it possible to replace the for-loop in Algorithm 2 with another loop? Argue either why not or give an example how to replace it (write the algorithm anew).

Bonus exercise:
k) *Ponder and compare* the memory consumption of Algorithm 1 and 2.

2) Terminology (4 x (1p +1p) = 8p)

Define the following *concepts* (4 x 1p). In addition, *give an example* of each (4 x 1p).

a) *Abstract Data Type* (ADT)

b) *Priority Queue*

c) *Binary Heap*

d) *Heap Property*

3) Mergesort (2p + 2p + 2p + 2p + 2p = 10p)

a) *Describe* how *merging* works. *Illustrate* your explanation with appropriate drawings.

b) *Argue* whether it is true or false: merging is more suitable for *arrays* than *linked lists*. Hint: there is no true/false answer to this question. Try to find out at least two different viewpoints to this question. The points are given based on feasible argumentation only.

c) *Describe* how *mergesort* works. *Illustrate* your explanation with appropriate drawings.

d) Analyze the worst case time complexity of mergesort based on your description.

e) Consider the following list of items: 3, 1, 6, 2, 5, 7, 4, 8, 9, 0. Show step by step how mergesort sorts the list.

4) Search Structures (2p + 2p + 2p + 4p = 10p)

a) *Define* the concept *Search Structure*. *Give examples* of search structures.

b) *Describe* the fundamental principles of *Balanced Search Trees*.

c) *Describe* the fundamental principles of *Hashing*.

d) *Analyze* the time complexities of fundamental search structure operations for Balances Search Trees (BSTs) and Hashing. *Compare* BSTs with hashing based on worst case and average case analysis of different operations.

6) Time on Examination (0p)

Estimate the time you used for answering the exam with 15 minutes precision.

5) Feedback (2p)

Provide course feedback due December 14th with the form that you can find in the Noppa page.