

Algorithms and Data Structures, Reexam March 2011
IT University of Copenhagen

Instructions

You will draw two sheets with questions — one of type A and one of type B. Use the preparation time to prepare yourself to discuss the questions you have drawn in the oral exam. *All written aids are allowed.*

You may take notes that you bring to the oral exam. We will use up to 15 minutes discussing these questions, after which we will be asking questions regarding other aspects of the course and/or the project. It is your responsibility to use this time to show your competences within the course related to the questions (i.e., “making the time pass” is a losing strategy).

Runtime analysis

Suppose that A is an array that contains n **distinct** values. Analyze the asymptotic worst-case running time of the pseudocode below (function of n).

```
1: Let  $T$  be an empty binary search tree storing triples of integers
2: for  $i := 1$  to  $n$  do
3:   for  $j := 1$  to  $n$  do
4:     for  $k := 1$  to  $n$  do
5:       if  $A[i]*A[j]==A[k]$  then
6:         insert  $(i, j, k)$  into  $T$ 
7:       end if
8:     end for
9:   end for
10: end for
```

Graph algorithms

For software that analyzes social networks, you want to be able to find the “centrality” of a person in the network. You can think about the network as a graph where people are vertices, and edges indicate friendships. Assuming the graph is connected, then for any two persons p and p' there is some shortest sequence of persons p_1, \dots, p_k such that p is connected to p_1 , p_1 is connected to p_2, \dots , and p_k is connected to p' . The question we are interested in is:

For a given person p , how far away is the farthest person in the social network?

You are given a file of the relevant information, where the line:

Rasmus : Jesper

means that the users with user names **Rasmus** and **Jesper** are friends. Suggest a data structure for representing the friend information, and an algorithm for computing the distance to the farthest person. You should not design a new algorithm, but rely on the toolbox you have seen in the course.