Final exam in IT1105/TDT4120 Algorithms and Data Structures

Exam date	Thursday December 6
Exam time	1500–1900
Grading date	Sunday January 6
Language	English
Contact during exam	Magnus Lie Hetland (ph. 91851949)
Aids allowed	All printed/handwritten; specific, simple calculator

Please read the entire exam before you begin, use your time wisely, and prepare any questions before the lecturer arrives. Make and state assumptions where necessary. **Please write short and concise answers.** Long explanations that do not directly answer the problems will be given little or no consideration. Preferably write where indicated (i.e., without submitting extra sheets).

Answer form for problem 1

(See problem text on page 2. Wrong = 0 points, no answer = 1 point, correct = 4 points.)

	Yes	No	Brief reasoning
a	☐	☐	
b	☐	☐	
c	☐	☐	
d	☐	☐	
e	☐	☐	
f	☐	☐	
g	☐	☐	
h	☐	☐	

Answer form for problem 2

(See problem text on page 3. Wrong = 0 points, no answer = ¼ points, correct = 1 point.)

Recursive calls	Size, subproblems	Work in each call	Recurrence	Running time
One	Reduced by 1	Constant	$T(n) =$	$\Theta($ $)$
One	Halved	Constant	$T(n) =$	$\Theta($ $)$
One	Reduced by 1	Linear	$T(n) =$	$\Theta($ $)$
One	Halved	Linear	$T(n) =$	$\Theta($ $)$
Two	Reduced by 1	Constant	$T(n) =$	$\Theta($ $)$
Two	Halved	Constant	$T(n) =$	$\Theta($ $)$
Two	Reduced by 1	Linear	$T(n) =$	$\Theta($ $)$
Two	Halved	Linear	$T(n) =$	$\Theta($ $)$

Please answer problems 3 through 5 in boxes in the text.

Problem 1 (32%)

For each subproblem below, please check either «Yes» or «No», and very briefly state your reasoning. Please use the answer form on page 1.

Note: This problem is scored as follows: Wrong answers or wrong reasoning for a subproblem gives 0 points, unanswered subproblems give 1 point, and correct answer (including reasoning), 4 points. In other words, an evaluation of our own knowledge is awarded – if you are less than 25% certain of your answer (including reasoning) it pays to leave the answer blank.

a. It is easier to find the edge out-degree of a node if we use adjacency matrix representation than if we use adjacency lists.

b. How we implement Quicksort has great impact on how vulnerable it is to unfavorable data sets.

c. Assume that you sort n real numbers and then extract 10 numbers with indexes spread as evenly as possible from 1 to n. Could you have found the same elements without sorting, in linear time?

d. Because of the UNION/FIND-SET-mechanism, Kruskal's algorithm can find minimal spanning trees for disconnected graphs as well as connected.

e. The basic method in the curriculum for finding maximum flow has an exponential worst case running time.

f. Some NP-complete problems involve finding an optimal permutation of n elements. It is possible that some of these problems have a worst case

running time of $\Omega(n!)$

g. An algorithm takes the integer n as its only input and uses binary search to find an integer value in the interval $[1, n]$ that is optimal (in some sense). Assume that each operation of the binary search takes constant time and that the binary search dominates the total running time. The algorithm has linear running time.

h. Building a heap has the same expected, asymptotic running time as building a search tree.

Problem 2 (8%)

In the answer form on page 2 you see a table of running times for a set of recursive algorithms. Fill in what's missing. Assume $T(0) = T(1) = \Theta(1)$.

Note: This problem is graded as follows, for each row as a whole: Wrong answers give 0 points, missing answers give ¼ points and correct answer (on both questions), 1 point. In other words, an evaluation of our own knowledge is awarded – if you are less than 25% certain of your answer for a row it pays to leave the answer blank.

Problem 3 (27%)

Assume that you are to implement a data structure that lets you keep track of two things: (1) The order in which elements are added and removed, and (2) whether an elements is to be found in the structure. In other words, you want a structure that can work both as a FIFO queue and a set.

a. How would you implement such a structure? Emphasize asymptotic running time. What is the running time for the different operations? (Use Θ notation and briefly state your reasoning.)

Answer (12%):

Assume you have a map of an archipelago in the form of a grid where each square is either sea/lake or land (assume that all edge squares are sea). You wish to write an algorithm with linear running time that counts how many islands are found on the map.

Note: Each separate, connected segment of land counts as an island. In other words, islands in lakes on other islands are to be counted.

b. Briefly describe such an algorithm.

Answer (8%):

You have constructed a recursive algorithm that finds both the maximum and the minimum in an array. It works by recursively finding the maximum and minimum in each half, and then picking the maximum and minimum from each of the two answers. Assume that the size of the array is $n = 2^k$, for an integer k.

c. How many comparisons will your algorithm use? State your reasoning.
Hint: The answer is not n–1 or $2n$–2.

Answer (7%):

Problem 4 (7%)

Consider the following algorithm.

```
DOLOREM(ipsum, quia, dolor)
amet ← ipsum[dolor]
consectetur ← quia–1
for adipisci ← quia to dolor–1
    if ipsum[adipisci] ≤ amet:
        consectetur ← consectetur + 1
        ipsum[adipisci] ↔ ipsum[consectetur]
ipsum[consectetur+1] ↔ ipsum[dolor]
return consectetur+1

NUNQUAM(ipsum, quia, dolor, velit)
sit ← DOLOREM(ipsum, quia, dolor, quia)
if sit < velit:
    NUNQUAM(ipsum, velit, sit+1, dolor)
else if sit > velit:
```

NUNQUAM(*ipsum*, *velit*, *quia*, *sit*–1)

a. What does the algorithm NUNQUAM do? (That is, what does it compute – what is its purpose? A summary of what steps it takes will give few or no points.) Assume that *ipsum* is a 1-indexed sequence. Make any other natural assumptions about the arguments to NUNQUAM.

Answer (7%):

Problem 5 (26%)

Assume that you have a local computer network where every machine is linked to the others, directly or indirectly. The cables you have used have varying prices (in whole kroner/NOK) and quality (a real number greater than or equal to 0 and less than 10). You cannot assume any direct relationship between price and quality.

You are to pick a set of cables for sale, and place the following requirements on the selection:

- All machines must be connected, directly or indirectly, after the chosen cables have been removed.
- All links remaining must have a quality of at least k.

Given these requirements you are to find a set of cables that gives you the highest possible sales sum.

a. Briefly describe an efficient algorithm that solves the problem in general.

Answer (12%):

After some thought you add a requirement:

- If there are more than one optimal cable sets, you are to choose among them so that the remaining cables have the highest possible total quality.

In other words you are still to find a set that gives you the highest possible sales sum, that doesn't break up the network, and that lets each remaining cable have

a quality of at least k. Letting the remaining network have the highest possible total quality is an additional requirement.

b. Describe briefly an efficient algorithm that solves the problem in general.

Answer (7%):

You have decided to use operating systems from a specific, commercial vendor. You have bought a license (valid for one machine) for each version of their newest operating system. Your goal is to have each your computers running one of a set of server programs you have written. The problem is that (1) not all operating system versions can run on all machines, and (2) not all of your server programs can run on all the operating system versions.

c. Briefly describe an efficient algorithm that decides whether it is possible to run all your server programs (assume that it is not possible to run two server programs on the same machine).

Answer (7%):