TDT4120 Algorithms and Data Structures

Examination, December 12, 2023, 15:00–19:00

Academic contact Magnus Lie Hetland

Support material code E

Problems

1 In the following subtasks, your answer is meant to be *very brief*.

5 % *a*) What is the best-case running time of Insertion-Sort?

5 % *b*) What is the greatest advantage of an array over a linked list?

5 % *c*) In Counting-Sort(A, n, k), A is an array with n values. What is k?

5 % *d*) What is topological sorting?

5 % *e*) In a max-heap, the value x is in a parent node, while the values y and z are in the left and right child nodes, respectively. What requirements are placed on the relationship between the values?

Here we are asking about the *values*, not indices, for example.

2 In the following subtasks, information is given about the functions $f(n)$ and $g(n)$. In each case, express $f(n) + g(n)$ in asymptotic notation.

During the exam, the following clarification was given: The phrase "in each case" means "for each subtask." Each subtask is to be answered with one expression.

5 % *a*) $f(n) = O(n^2)$, $f(n) = \Omega(n)$, $g(n) = O(n^2)$, $g(n) = \Omega(n^2)$

5 % *b*) $f(n) = \Omega(n^2)$, $f(n) = \omega(n)$, $g(n) = O(n^2)$, $g(n) = o(n^3)$

3 Solve the following recurrences. Give your answer in Θ-notation.

5 % *a*) $T(n) = T(n-1) + n^2 - (n-1)^2$

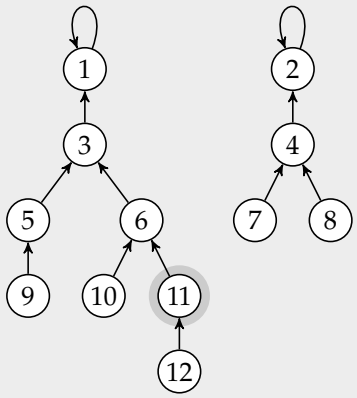
5 % *b*) $T(n) = 2T(n/4) + \sqrt{n}\lg^2 n$

5 % **4** You are given the following frequencies for the alphabet `a`, . . . , `h`:

`a`:1 `b`:1 `c`:2 `d`:3 `e`:5 `f`:8 `g`:13 `h`:21

In a Huffman code for these characters, what is the number of digits for the character `e`?

Figure 1



Here we are looking for the number of binary digits needed to encode one `e` with the Huffman code, not the total number of digits used for all the `e` characters in the text.

5 % **5** In Figure 1 you see a disjoint-set forest. We can represent the parent pointers with a table A, where $A[v] = v.p$:

$A = \langle 1, 2, 1, 2, 3, 3, 4, 4, 5, 6, 6, 11 \rangle$

Execute FIND-SET(11) and update A. What does A look like afterwards?

Answer by listing the numbers in A. You don't need to write $A = \langle \ldots \rangle$.

5 % **6** Which augmenting paths will EDMONDS-KARP find in the flow network in Figure 2?

Here you are supposed to perform EDMONDS-KARP on the flow network, but without the initialization $(u, v).f = 0$. Give the paths as sequences of nodes. Write one path per line, in the order in which they occur. For example:

1, 2, 3, 4, 5
7, 6, 5, 4, 3, 2, 1
4, 5, 6, 7, 8, 9

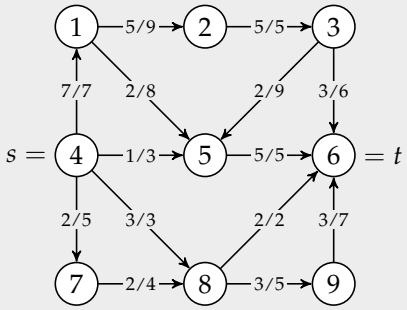
(This is only an example of the format, not an actual valid set of paths.)

5 % **7** If there is a tie between some candidates in some of the rankings in the stable-marriage problem, is it still certain that we can find a stable matching? Explain briefly.

8 Consider the algorithm UNTITLED (Algorithm 1), where $A[1:n]$ is an array of integers and the algorithm is started with the call UNTITLED$(A, 1, n, k)$.

5 % *a*) What does the algorithm do?

Figure 2



Algorithm 1

```
UNTITLED(A, p, r, k)
1  if p < r
2      q = RANDOMIZED-PARTITION(A, p, r)
3      UNTITLED(A, p, q − 1, k)
4      if q < k
5          UNTITLED(A, q + 1, r, k)
```

Here we are looking for the result of running the algorithm, or what problem it solves, not how it behaves, step by step.

5 % *b*) What is the expected running time, as a function of n and k? Give your answer in Θ-notation.

As a simplification, you can assume that q always ends up midway between p and r.

9 You are to create a playlist that is exactly t seconds long. You have n songs to choose from. You may assume that every song lasts a whole number of seconds.

5 % *a*) How can you show that this is a hard problem?

5 % *b*) How can you solve the problem?

5 % **10** Your friend Smartnes is looking for paths in a connected weighted undirected graph, from a source vertex s to all others. If he finds the shortest paths, the sum of all path lengths will be as small as possible, but the parts of the paths where they overlap will then be counted multiple times. He would rather find a set of paths that minimizes a similar sum, where the overlapping parts are counted only once. How can he do that? Explain briefly.

5 % **11** Your friend Klokland is responsible for a concert series, but due to budget cuts, she has to make do with just one stage. Several of the concerts overlap in time,

and some therefore have to be canceled. Klokland wants to cancel as few as possible.

All she has kept track of is the start times of all the concerts, and an undirected graph with concerts as nodes, and edges between those that clash. She wants to remove as few nodes as possible so that all the edges disappear.

She gets a bit sweaty as she realizes that this is the optimization version of the VERTEX-COVER problem, but she hopes maybe you have some good ideas.

Construct and describe an algorithm that solves the problem generally.

5 % **12** Construct and describe an algorithm that determines whether a directed graph has an odd cycle, that is, a cycle with an odd number of edges.

Here, you get full marks with a running time of $O(V^3)$.

Hint 1: It may be useful to look at paths as part of the solution.

Hint 2: You need not limit yourself to *simple* paths and cycles.

Hint 3: Is there an odd path from i to j? How about one with an even number of edges?