Final exam Algorithmics SS 2020

Iwanowski 02.09.2020

Hints:

Time limit: 120 minutes

Admitted appliances: none

Please give your answers and interim results exclusively in the pages of the assignments. If the space is not sufficient, you may use the blank reverse sheet on the opposite side.

Language: You may answer each assignment in German or English just as you feel most comfortable in order to express your thoughts and intentions clearly. In particular, you may also switch the language between or within the assignments.

This exam consists of 11 pages including this cover sheet.

This exam issues 50 evaluation credits (EC).
For passing this exam you need at least 25 EC.

Good luck!

Assignment 1: 7 EC

Compare the worst case behaviour of the algorithms Quicksort and Mergesort :

a) Define recursive inequalities for n input items respectively (one for Quicksort, one for Mergesort). (2 EC)

b) For both algorithms specify the asymptotic run time in n respectively. (2 EC)

c) Show the inductive step for the proof of Mergesort's run time:
First, specify what has to be proven for n.
Second, show the proof using the validity of your claim for numbers smaller than n.
For simplicity, you may assume that n is a power of 2. (3 EC)

Assignment 2: 2 EC

Binary search is already an optimum search algorithm for items in a sorted array.

a) Which feature should be improved by refined strategies such as interpolation search or quadratic binary search?

b) Is there also a deterioration by the application of these algorithms? If so, which one?

Hint for the expected answer: You need not (but may do) give actual run times. But you must mention what the run times refer to.

Assignment 3: 7 EC

Simulate the deterministic algorithm Select (k, A) with linear run time (worst case) for the following input:

k = 12, A = [13, 2, 19, 15, 1, 28, 29, 16, 11, 37, 4, 31, 17, 33, 3]

Specify the actual steps of the top level procedure and give the respective output for each step directly (not in general but with this example). You need not justify the results of these steps by showing the lower level invocations but should rather give these results directly (by a sharp look to this easy example).

Assignment 4: 4 EC

Consider the solution of the dictionary problem using a (2,3) tree:

a) Which are the invariants to be maintained in the data structure? Specify exactly where the keys and where the values are stored. (2 EC)

b) Given an inner node of a (2,3) tree (not being the root): Specify the interface of the insert method in words. (2 EC)

Hint for the expected answer: Do not show the answer with an example. Give the answer in general (you may use your own words).

Assignment 5: 6 EC

Analyze Bellman's algorithm given by the following code:

The input is an array [a_1, a_2, ..., a_n].
For each i, p_i is the probability that an item equal to a_i is asked for, and q_i is the probability that an item between a_i and a_{i+1} is asked for (with natural extensions for the border cases 0 and n).

Algorithm 3: [Bellman, 1957] Iterative Suche nach dem optimalen Suchbaum T.

```
for i = 0, ..., n do
    w_{i+1,i} = q_i
    c_{i+1,i} = 0
end for
for k = 0, ..., n − 1 do
    for i = 1, ..., n − k do
        j = i + k
        Determine m where i ≤ m ≤ j s. that c_{i,m−1} + c_{m+1,j} is minimal
        r_{i,j} = m
        w_{i,j} = w_{i,m−1} + w_{m+1,j} + p_m
        c_{i,j} = c_{i,m−1} + c_{m+1,j} + w_{i,j}
    end for
end for
```

a) Which problem does this algorithm solve? Do not answer with "optimum search tree" but determine what this means. (1 EC)

b) What do the variables r_{ij}, w_{ij}, c_{ij} stand for?
(Hint: First define the subtree T_{ij} before you specify the meaning of the variables). (2 EC)

c) Specify the asymptotic run time of this algorithm and justify this by giving the run time of the individual parts of this program (you may enter the justification in the code above). (3 EC)

Assignment 6: 7 EC

Analyze Kruskal's algorithm for computing the minimum spanning tree in a graph with n nodes and m edges:

a) Describe how it works in general by giving the decisive steps of this algorithm. (4 EC)

b) Denote the asymptotic run time for the steps described in a) and derive the run time of the total algorithm. (3 EC)

Assignment 7: 4 EC

Consider the problem to determine the maximum flow in a network with given flow capacities for the edges.

a) Define what is an augmenting path and specify the relation to the maximum flow. (2 EC)

b) Prove at least one part of the relation defined in a)
(Hint: First consider which is the easy part and prove only this one) (2 EC)

.

Assignment 8: 4 EC

a) Define the input and output of the problem to find the maximum matching in a graph. In particular, explain what “matching” means for the edges. (1 EC)

b) Sketch a construction which reduces the problem of finding the maximum matching in a **bipartite** graph to the problem of finding a maximum flow in a flow network. Your construction should specify the network with its significant properties. You **should** illustrate your explanation with a graphical sketch. (3 EC)

Assignment 9: 7 EC

a) Sketch how the problem Convex Hull **in the plane** is solved in a trivial way. Specify and justify the asymptotic run time. (3 EC)

b) Sketch how Convex Hull in the plane may be solved when the Voronoi diagram is already computed. Specify and justify the asymptotic run time now. (3 EC)

c) Is it worth to compute the Voronoi diagram first when only Convex Hull has to be computed? Justify your answer with the specification of run times. (1 EC)

Assignment 10: 2 EC

Consider Fortune's algorithm for the construction of a Voronoi diagram via plane sweep:

Normally, a plane sweep works with two parallel lines sweeping over the plane. For the Voronoi algorithm, one of these lines must be substituted by something else. Describe what it must be substituted by. You may additionally illustrate this with a sketch but should also describe it in words.