Resit B Data Structures and Algorithms 2019-2020

Monday July 6, 09.00-11.00

4 exercises

Remarks

- The schedule for the exam is as follows:
  08.30–09.00: id-check
  09.00–11.00: work on exam
  11.00–11.10: submit your work via Canvas
- You work with pen and paper only.
- During the exam, communication with others is forbidden, so:
  mobile phone is off or silent,
  mobile phone can be used as camera,
  no headphones or ear pods.
- If you need to go the the bathroom, ask permission via the chat.

Statement of integrity:

I hereby declare, that I understand that taking an online exam during this corona crisis is an emergency measure to prevent study delays as much as possible. I know that fraud control will be tightened and I also realize that a special appeal is now being made to trust my integrity. With this statement, I promise to:

- make this exam completely on my own,
- do not consult sources, other than explicitly mentioned in the examination,
- make myself available for any oral explanation of my answers.

Exercise 1. (*10+10 points*)
This exercise is concerned with sorting. The pseudo-code for partition is given:

```
Algorithm partition(A, p, r):
  x := A[r]
  i := p − 1
  for j = p to r − 1 do
    if A[j] ≤ x then
      i := i + 1
      exchange A[i] with A[j]
  exchange A[i + 1] with A[r]
  return i + 1
```

(a) Apply partition to the array $A = [5, 1, 4, 6, 2, 4]$.

Consider every iteration of the for-loop, and indicate in a picture the array A and the indiced i and j.

(b) Solve the recurrence equation for the worst-case time complexity of quick-sort:

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n-1) + n & \text{if } n > 1 \end{cases}$$

Exercise 2. (*10+10+10 points*)
This exercise is concerned with data structures.

(a) Describe how to implement a queue using a singly linked list. Give a procedure for adding an element and a procedure for deleting an element; if necessary test on overflow or underflow.

(b) Construct an AVL-tree by successively inserting the numbers

7 9 14 5 16 2 1

starting from the empty tree. Restructure if needed.

Once the complete AVL-tree has been constructed: remove the root of the tree; again restructure if needed.

(c) Apply 'on the fly' the algorithm for bottom-up max-heap construction to the array $A = [1, 2, 3, 4, 5, 6, 7]$; give your answer in pictures.

Exercise 3. (*10+10 points*)
This exercise is concerned with greedy and dynamic programming.

(a) Consider the algorithm for a longest common subsequence (LCS) of input sequences $X = \langle x_1, \ldots, x_m \rangle$ and $Y = \langle y_1, \ldots, y_n \rangle$:

```
Algorithm LCS(X, Y):
  new array C[0 ... m, 0 ... n]
  for i := 0 to m do
    C[i, 0] := 0
  for j := 0 to n do
    C[0, j] := 0
  for i := 1 to m do
    for j := 1 to n do
      if x_i = y_j then
        C[i, j] := C[i − 1, j − 1] + 1
      else
        C[i, j] := max(C[i, j − 1], C[i − 1, j]
  return C
```

Apply the algorithm to the following input: $X = \langle A, B, C, D, E, F \rangle$ and $Y = \langle A, D, F, C, B, D \rangle$. Give your answer in the form of a table and give explicitly the longest common subsequence(s) that is (are) found.

(b) Consider the making change problem: Suppose that we have available coins with postive integer values $c_1, c_2, \ldots, c_n$, in increasing order with $c_1 = 1$, and suppose that we have an unlimited amount of every coin value available. The question is how to make value n with a mininum number of coins.

Describe (in pseudo-code) a greedy algorithm MakeChange(n) that computes this minimum.

Is your algorithm correct for any choice of $c_1, c_2, \ldots, c_n$, with $c_1 = 1$? Why (not)?

Exercise 4. *(10+10 points)*
This exercise is concerned with various subjects.

(a) Suppose we have available a priority queue, with operations for adding and deleting. Describe (in a precies way, but not necessarily in pseudo-code) how we can use the priority queue to give a sorting algorithm.

(b) What is the worst-case time complexity of searching for an element in a hash table of size m, containing n elements, where collision is solved by chaining? Explain your answer.

The mark for the exam is (the total number of points plus 10) divided by 10.