Algorithms and data structures 1

Theoretical exam

1. Night appointment

July 8, 2022

Surname:	
Student number:	

The information is printed on both sides!

7	11	23	21	17	20	35	10
+	+	+	+	+	+	+	+

Task 1 [2]

In the table above, enter the digits of your student number in the empty boxes in front of which there is a plus sign.
Do the additions and find the numbers up to .

Task 2 [18]

a) [13] In C++-like pseudocode, create **a** recursive function f with an integer parameter n to which the master theorem is applicable and whose runtime complexity is **simultaneously** in ÿ(Parameter ***b*** in the master theorem must take the value % 2+ 3. ^{3} log), ÿ(log) and O(^{5}) lies. The

2 b) [5] Using the master theorem, show that f has the desired runtime complexity.

Task 3 [20]

The values up to · (from task 1) are stored in an array in this order from left to right. Sort the values in ascending order

a) [10] Quicksort
b) [6] Heapsort
c) [4] Merge sort

Specify all the necessary steps in sufficient detail to make it clear how the algorithm works.

Task 4 [20]

a) [8] Insert the values up to from task 1 (in this order) into an initially empty binary search tree.
Sketch the state of the tree after each insertion step.
Note: the tree can contain multiple values.

b) [3] In notation similar to C++, give the definition of a data structure for a binary search tree.

c) [5] In notation similar to C++, give a definition of a function that traverses the depth first binary search tree and returns all stored values.

d) [4] Record the output of your function when it is called on the tree created in **subtask a) .**
Is this a preorder traversal, a postorder traversal, an inorder traversal or another type of traversal?

Task 5 [20]

a) [10] Add the numbers until (from example 1) in this order into an originally empty hash table of size 7

a. Use the hash function ( ) = % and double hashing with ( ) = % + as the second hash function for collision handling. Indicate the state of the hash table after each insert operation.

b) [4] Specify the collision path that will be searched when attempting to in the hash table populated after a). additionally (from example 1).

c) [2] Why is the “recoverable” mark used in double hashing?

d) [2] Why is it recommended to use a table size that is a prime number for double hashing?

e) [2] What is the fundamental difference between static and dynamic hashing methods?

Task 6 [20]

The following adjacency matrix is given, which represents the costs of the connections between the nodes of a directed graph describes (from example 1):

0 58 0 2 3 0 0
6   4

8th
0 0 0 7
2
0 1   0
3   7
(42 2 0   0)
5

a) [2] Sketch the graph described by this adjacency matrix.

b) [10] Use Dijkstra's algorithm to determine the shortest paths from node 1 to all other nodes of the Graph (where node 1 corresponds to the node of the first row/column in the adjacency matrix).

c) [8] Use Prim's algorithm to determine a minimally spanning tree of the shadow of the graph. (She obtain the shadow of the graph by neglecting the directions of the edges. If two nodes are then connected by two or more edges, these edges are combined into one. In other words: Two nodes x and y in the shadow are connected by an undirected edge if and only if at least one of the edges from x to y or from y to x exists in the originally directed graph. As the weight of the undirected edge, choose the minimum of all directed edges represented by it.) Note all intermediate steps so precisely that it is clear when which edge is added to the spanning tree.