1.	
2.	
3.	
Sum	

Re-exam

Algorithms and Data Structures (Dat 1 and SW 3)

29.02.08

Finn V. Jensen

Full name	
CPR number	
e-mail at cs.auc.dk	

Read first

- Start by filling in your name, CPR number, and e-mail address. Remember to put your name and CPR number also on any additional sheet of paper you will use.
- The exam consists of three questions. All questions are relative to the topics covered during the lectures and to the content of the main textbook by Thomas H. Corman et al. You have 3 hours to answer the questions. It is recommended to spend approximately one hour per question. Make sure that your writing font is readable.
  - In Question 1 mark the correct answer out of the offered choices, or write the result directly into the highlighted box. For any calculations you need to do use a separate sheet of paper (this paper is not a part of the answer, do not hand it in). Do not provide any additional justification of your answers in this question.
  - In Questions 2 and 3 read carefully what you are asked to do. Make sure that you answer all the subquestions, and only those. Your answers should be clear and precise. Whenever you are asked to write an algorithm, use your favorite pseudo-code (the one used during the lectures is recommended but not required). It is also worth to write two or three lines describing informally what the algorithm is supposed to do. Whenever using big O notation to analyze the complexity of algorithms, always give the best possible estimate.
- It is not allowed to use any electronic devices like laptops, mobile phones, and I-pots except for a basic non-programmable calculator (though it will not be needed). You can freely use your notes from the lectures, textbooks, or any other course material.

Question 1 (*In total 50 points*)

a) (*6 points*)
Mark the correct answer by putting a cross in the corresponding field.

$\frac{n^5}{2048} + 1024n^2 + 2n^{-5}$ is	☐ $\Theta(n^5)$	☐ $\Theta(n^2)$	☐ $\Theta(n^{-5})$
$\log(n^2 \cdot 2^n) + 1022n$ is	☐ $\Theta(n)$	☐ $\Theta(n \log n)$	☐ $\Theta(2^n)$
$(\log n)^n + n^{100} + 2^n$ is	☐ $\Theta((\log n)^n)$	☐ $\Theta(n^{100})$	☐ $\Theta(2^n)$

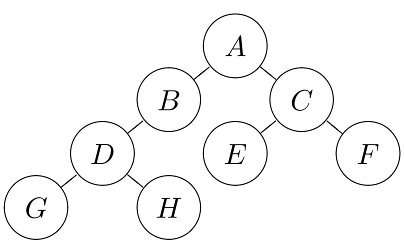
b) (*8 points*)
Consider the priority queu below, organized as a max-heap.

1	2	3	4	5	6	7
6	5	4	3	2	1	

Write in the box below the result of first inserting "7" (MAX-HEAP-INSERT) end then removing "7" (HEAP-EXTRACT-MAX).

1	2	3	4	5	6	7

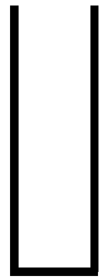
c) (*7 points*)
Write the sequence of visited nodes in a postorder traversal of the binary tree.



, , , , , , ,

d) (*6 points*)
Fill in the stack s below with values resulting from the following operations.

s:STACK; q:QUEUE; s.make; q.make
q.enqueue(1)
q.enqueue(3)
s.push(q.dequeue)
s.push(4)
q.enqueue(s.pop)
s.push(q.dequeue)



e) (*7 points*)
Given the hash function $h(k, i) = (k + \frac{1}{2}i^2 + \frac{1}{2}i)\ mod\ 8$, and the key sequence 8, 7, 17, 11, 16. Insert all keys into the hash table $A[0..7]$ below. Use quadratic probing to resolve collisions.

0	1	2	3	4	5	6	7

f) (*8 points*)

Consider the following piece of code

$i \leftarrow 1$
$k \leftarrow 1$
while $i < n + 1$ **do**
    $k \leftarrow 3k + 1$
    $i \leftarrow i + 1$

Which of the invariants below holds for the loop?

☐ $k = i^2$ ☐ $k = i^3 - 3i^2 + 5i - 2$ ☐ $k = \frac{1}{2}(3^i - 1)$ ☐ $k = 3i - 2$

g) (*8 points*)
Consider the following recurrence equation (for simplicity, assume that n is a power of 2):

$T(1) = 1$
$T(n) = 8 \cdot T(\frac{n}{2}) + n^3$ for $n \geq 2$.

$T(n)$ is
☐ $\Theta(n^2)$ ☐ $\Theta(n^2 \log n)$ ☐ $\Theta(n^3)$ ☐ $\Theta(n^3 \log n)$

Question 2 *(25 points)*

When producing an item (like a ship) you have a partial temporal order of the jobs. A production assignment can be represented by a table as follows:

Job	A	B	C	D	E	F	G	H	I	J
Duration	2	3	1	4	2	3	5	1	2	1
End before	$\{F, I\}$	$\emptyset$	$\{E, G\}$	$\{A, H, I\}$	$\{B, J\}$	$\emptyset$	$\{E\}$	$\{B, I\}$	$\{B, F\}$	$\emptyset$

For example, the table expresses that job A takes 2 units of time, and it must be finished before jobs F and I can be dealt with.
A similar phenomenon appears when running (object oriented) computer programs.
Note that the temporal requirement is transitive: if X must be finished before Y, and Y must be finished before Z, then X must be finished before Z.

a) Represent the problem as a graph, and provide a possible sequencing of the jobs when performed by one person.

The *size* of the representation is the number of jobs referred in the rows 'Job' and 'End before' (for the representation above, the size is 24).

b) Design an $O(size)$ algorithm to check that a given production assignment does not contain a deadlock: job X awaits Y to be finished, which awaits job Z to be finished,. . . , which awaits job X to be finished.

c) Assume that there is only one person to do the jobs. Design an $O(size)$ algorithm for establishing a sequencing of the jobs for this person.

d) Assume that there are as many persons available as may be requested. For the assignment above, what is the minimal time required for having all jobs done?

e) Describe an $O(size)$ algorithm that given an assignment representation calculates the minimal time required for doing all the jobs when you have all the staff you may need (Hint: you may modify the graph representing the problem)

Question 3 *(25 points)*

You have six cities A, B, C, D, E, F, and you have the following roads (distance indicated above the edge): $(A \overset{2}{-} B), (A \overset{3}{-} C), (A \overset{2}{-} F), (B \overset{2}{-} C), (B \overset{6}{-} D), (B \overset{3}{-} E), (B \overset{2}{-} F)(C \overset{3}{-} E), (D \overset{4}{-} E)$.

You wish to go shortest possible from A to D.

a) Show how to use Dijkstras algorithm to solve the problem. Draw the graph, and write the intermediate results in terms of the evolution of the set S of processed nodes after each iteration.

We assume that the roads are straight such that the distance is equal to the air-line distance. To help your search you know the air-line distances where there are no roads: from A to D the distance is 7, from F to D the distance is 8, and from C to D the distance is 6.

b) The shortest distance from A to B is 2, and from A to C it is 3. Why can I with the help of the air-line distances conclude that no path from A through C to D can be shorter than the path $A - B - D$?

This can be utilized by adding a heuristic to Dijkstra's algorithm: instead of successively extracting from the priority qeue a city of minimal distance from A you extract a city which looks most promising. For each node N in the priority qeue, add the air-line distance from N to the goal to the calculated road distance from the origin; extract a node of minimal sum.

c) Show how this modified algorithm solves the problem above. Write the intermediate results in terms of the evolution of the priority qeue Q and the set S of processed nodes after each iteration.

d) Write the approach from c) in a general algorithm design: it takes as input a graph representing a road map (with lengths), two cities, X and Y, and air-line distances for all pairs (Z, Y) without a connecting road. It returns the minimal road distance between X and Y.

e) Show that when a node N is extracted from the priority queue and placed in S, then the minimal distance from X to N is determined. (Hint: use the reasoning from b) and the triangular inequality to show that all nodes on a minimal path from X to N are extracted before N).