Faculty of Science
Final Examination

Computer Science 308-250B section I
Introduction to Computer Science

Examiner: Prof. Claude Crépeau **Date:** April 28, 2000
Associate Examiner: Prof. Nathan Freidman **Time:** 9:00 – 12:00

INSTRUCTIONS:

- This examination is worth 60% of your final grade.
- The total of all (sub-)questions is 65 points.
- Each (sub-)question is assigned a value found in brackets next to it.
- O P E N • B O O K S •/• O P E N • N O T E S
- Faculty standard calculator permitted only.
- This examination consists of 3 pages including title page.
- This examination consists of 6.

SUGGESTION : read all the questions and their values before you start answering.

[10pts]

1) Consider the following pattern: "barbados".

- Construct the failure function of the Knuth-Morris-Pratt algorithm for this pattern.
- Draw a FSA that accepts exactly this pattern.

[10pts]

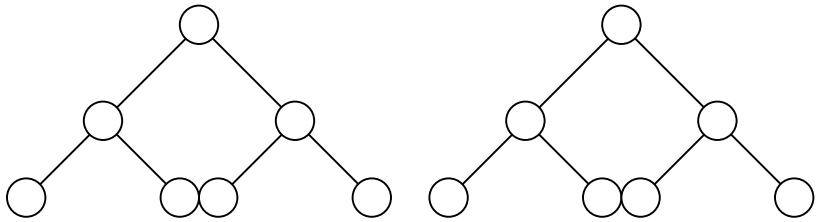
2) HEAPS

a) What is the largest number of disitnct integers that we can put in a binary tree that satisfy both the definitions (explain your answers)

- of a **(min)-heap** and of a **binary search tree** ?
- of a **(max)-heap** and of a **binary search tree** ?

b) Consider the following list of numbers : { 3, 19, 17, 11, 7, 9, 21 }.
Give two distinct ways of putting these elements in a (max)-heap.

Reproduce these two trees in your exam book and fill them in with the appropriate numbers to form two distinct (max)-heaps.

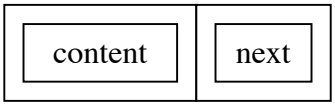


[10pts]

3) Consider the following (**Java and graphical**) definition of a node

```
Class node
{
Object content;
node next;
}
```

node:

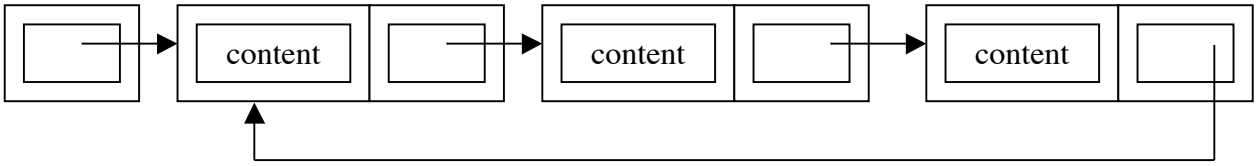


Let **h** be an object of type node, the head of a linked list of nodes.

Give the pseudo-code description of an algorithm "reverse(node **h**)" that reverses the nodes in the list without actually moving the contents.

Make sure that your algorithm works even if the list loops on itself at the start, such as :

h :



[10pts]

4) For each statement, say if it is *true* or *false*.

Right = +2pts, Wrong = -1pt, Blank = 0pt.

(a) In order to compute the fibonacci number f_n, any algorithm requires at least time $\Omega(n)$.

(b) The regular expressions **a(b+a)*** and **a+a(a+b)*** generate the same strings.

(c) When you calculate a **simple closed path** using the algorithm seen in class around the vertices of an n-sided polygon, you can determine in time O(log n) if a given point is a vertex of the polygon or not.

(d) The convex hull of a rectangle is always the rectangle itself.

(e) Every planar graph is 4-colorable.

[10pts]

5) Suppose you are given a (single) queue Q containing distinct integers.

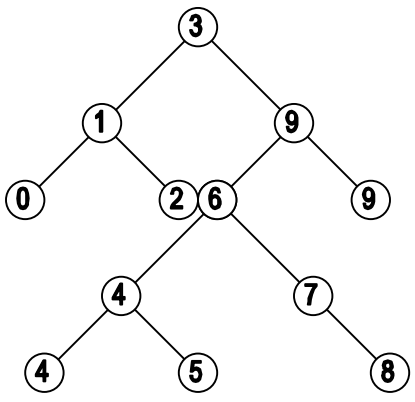
• Find an algorithm (in pseudo-code) to sort these numbers using nothing else than a fixed number of extra **int** variables (no array, no stack, no other queue) and queue operations on Q.

• Estimate the worst-case running time of your algorithm and express this time function using the Big-O notation.

[+5pts] • Argue that this would be impossible with a single stack S instead of a queue.

[10pts]

6) Consider the following binary tree. List the sequence of nodes visited according to a



• Depth-first search starting at (3).

• Breath-first search starting at (3).

Notice that this is a binary search tree.

• What is the successor of (2) ?

• Draw the resulting tree after removing (6).