CS 2223 D-Term 2020 SAMPLE MIDTERM EXAM Name:

Friday, April 17, 2020

Question 1 *(??? points)*
First Order Linear Recurrences

Solve the following recurrence relation using either the method of forward substitution or the method of backward substitution:

$W(n) = W(n-1) + 1, \quad$ for $n > 1; \quad W(1) = 1$

$W(n) = W(n-1) + n, \quad$ for $n > 1; \quad W(1) = 1$

$W(n) = 2W(n-1) + 1, \quad$ for $n > 1; \quad W(1) = 1$

$W(n) = 2W(n-1) + n, \quad$ for $n > 1; \quad W(1) = 1$

$W(n) = W(n/2) + n, \quad$ for $n > 1; \quad W(1) = n$, solve for $n = 2^k$

Question 2 *(??? points)*
Big O

QuickSort and MergeSort are both $O(n^2)$. However, of the two, only MergeSort is $O(n \lg n)$.

Explain why this is so.

Question 3 *(??? points)*
Master Theorem

Use the Master Theorem to find the running time of a Divide-and-Conquer algorithm that:

a. Divides a problem into *four* subproblems of *one quarter* size in constant time.

b. Divides a problem into *two* subproblems of *one half* size in linear time.

c. Divides a problem into *two* subproblems of *one half* size in quadratic time.

Question 4 *(??? points)* Graph Representations

Construct (visually) the graph that corresponds to the given matrix:

	1	2	3	4	5	6
1	0	0	1	0	1	0
2	0	0	0	1	0	1
3	1	0	0	1	1	0
4	0	1	1	0	0	1
5	1	0	1	0	0	0
6	0	1	0	1	0	0

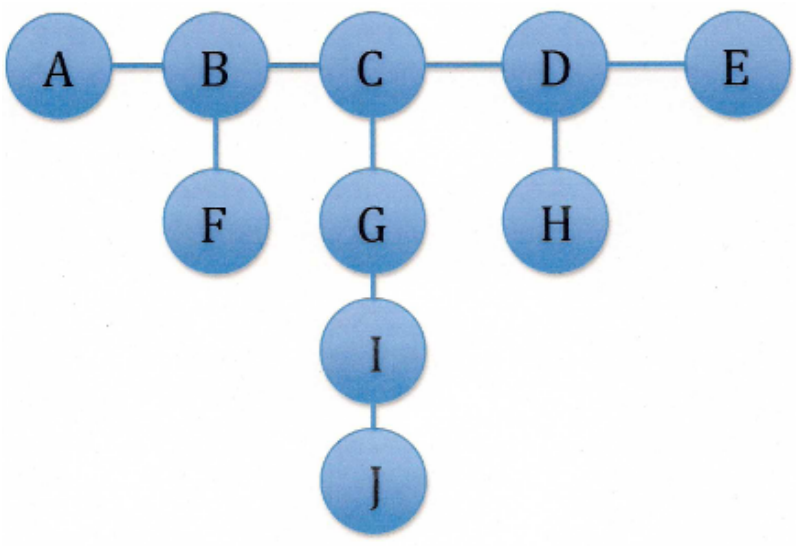
Also provide an adjacency list for the graph.

Is this a graph or a digraph? How can you tell without drawing it?

Question 5 *(??? points)*

Perform a traversal of the graph below by executing a Breadth-First search algorithm, starting at vertex A.
Report the vertices in the order they are discovered.

Perform a traversal of the graph below by executing a Depth-First search algorithm, starting at vertex A.
Report the vertices in the order they are discovered, i. e. "push" order.
Also, report the vertices in the order they are discarded, i. e. "pop" order.

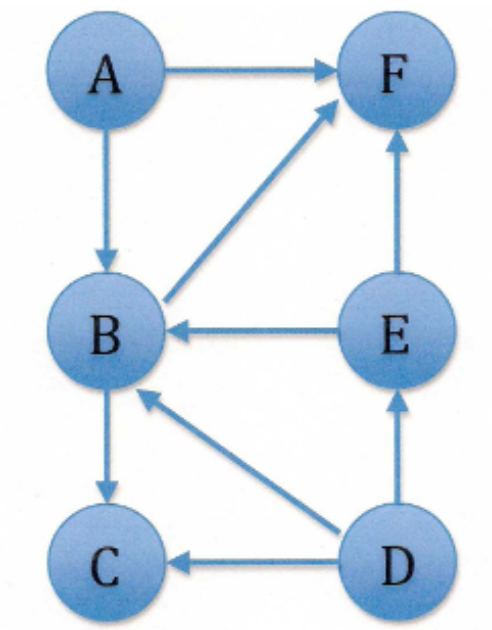


BFS order:

DFS push order of traversal:

DFS pop order of reporting:

Question 6 *(??? points)*
Perform a Topological Sort of the graph below. You may use the Source Removal Algorithm or the Depth-First Search algorithm.



Question 7 *(??? points)*
MergeSort or Quicksort

Trace the execution of the (ascending order) MergeSort algorithm on the following sequence:

5	8	7	4	6	1	2	3

Question 8 *(??? points)*
Heaps

Construct a heap from the numbers below

a) Using the bottom-up method, and

b) Using the top-down method.

(Solutions will assume a left-to-right sequence of insertions. Will right-to-left result in a different heap?)

5,10,12,8,7,3,4,13

Question 9 *(??? points)*
Horner's Rule

Evaluate $f(x) = 3x^5 + 4x^4 - 3x^2 + 2x - 1$ at $x = 2$ using Horner's Rule.

Note: You *must* show your work to receive credit, but you should also evaluate the polynomial in the traditional way in order to check your work.

Note also: There's a term missing. Don't forget to incorporate that into your evaluation. (Call it a "trick question", if you must.)

Note finally: Answers will be provided for $f(0), f(1), f(-1), f(-2)$, and $f(5)$ as well, so further practice will be rewarded. Of course, you can invent your own polynomial and test yourself.

Question 10 *(??? points)*
Use a dynamic programming algorithm to solve the coin row problem (maximize the sum without taking adjacent coins) for the following arrangement of coins:

3	2	1	6	4	7	8	5

Question 11 *(??? points)*
Read Pseudocode

Describe what the following algorithm does:

ALGORITHM *DoppelGänger*$(A[0..n-1])$
//??????????
//Input: (possibly empty) array A of arbitrary type
//Output: Boolean

tempFlag $\leftarrow$ False
$i \leftarrow 1$
while $n > 1$ **and** $i \leq= n-1$
    if $A[i-1] = A[i]$
        tempFlag $\leftarrow$ True
    $i \leftarrow i+1$
return tempFlag

Question 12 *(??? points)*
Write Pseudocode

Write pseudocode for an algorithm that examines array A of length n and returns "True" if it is sorted in ascending order and returns "False" otherwise. An array of length 0 should be considered sorted; you may NOT assume the array contains no duplicates.

Question 13 *(??? points)*
Sums

$\sum_{k=0}^{n} 2^k =$ a) $2^{n-1}-1$ b) $2^{n-1}+1$ c) $2^{n+1}-1$ d) $2^{n+1}+1$

$\sum_{k=1}^{n} k =$ a) n^2 b) $\frac{n(n-1)}{2}$ c) $\frac{n(n+1)}{2}$ d) $\frac{(n-1)(n+1)}{2}$

$\sum_{k=1}^{n} k^2 =$ a) n^3 b) $\frac{n(n+1)(2n+1)}{6}$ c) $\frac{n(n+1)(n+2)}{6}$ d) $\frac{(n+1)(n+2)(n+3)}{6}$

Question 14 *(??? points)*
Use the Euclidean algorithm to find the greatest common divisor of 210 and 308.