King Fahd University of Petroleum & Minerals

College of Computer Science and Engineering

Information and Computer Science Department

First Semester 111 (2011/2012)

ICS 202 – Data Structures

Final Exam

Tuesday, January 10th, 2012

Time: 120 minutes

Name: _____

ID# | | | | | | | | | |

Section 02 Dr. Wasfi 10-10:50am
Section 05 Dr. Sami 9-9:50am

Question #	Max Marks	Marks Obtained
1	20	
2	15	
3	25	
4	20	
5	20	
Total	100	

Instructions

1. Write your name and ID in the respective boxes above and circle your section.
2. This exam consists of 8 pages, including this page, plus one double-sided reference sheet, containing 5 questions.
3. You have to answer all 5 questions.
4. The exam is closed book and closed notes. No calculators or any helping aids are allowed.
5. Make sure you turn off your mobile phone and keep it in your pocket if you have one.
6. The questions are not equally weighed.
7. The maximum number of points for this exam is 100.
8. You have exactly 120 minutes to finish the exam.
9. Make sure your answers are readable.
10. If there is no space on the front of the page, feel free to use the back of the page. Make sure you indicate this in order not to miss grading it.

Q.1 [20 points] Multiple Choice Questions: Mark the best answer for each question below. Note: only one question should be chosen.

1. Consider the following code segment

```
sum = 0;
for (i=1; i<=n; i*=2)
  for (j=1; j<=i; j++)
    for (k=1; k<=5; k++)
       sum++; // Statement 1
```

The number of times Statement 1 is executed, assuming n is a power of 2, is equal to

a. n^2
b. $5n \log n$
c. $5 \log^2 n$
d. $10n - 5$
e. none of the above.

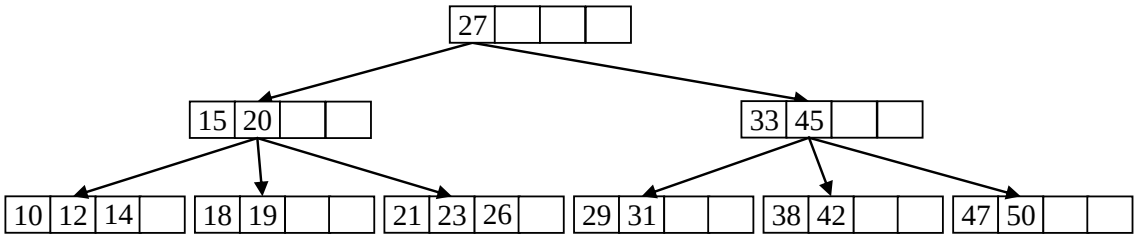
2. Consider the following method

```
1) public void final(int [] A, int n) {
2)   if (n ==0) {
3)     System.out.print(A[n]+" ");
4)     return;
5)   }
6)   final(A, n-1);
7)   System.out.print(A[n-1]+" ");
8)   final(A, n-1);
9) }
```

The number of times the print statements in lines 3 and 7 are executed is equal to

a. $2n - 1$
b. $2^{n+1} - 1$
c. $2^n - 1$
d. $\log (n+1) + 1$
e. none of the above.

3. Consider the following B-Tree.



The resulting tree will definitely have one level less than the current tree after deleting the key

a. 27
b. 18
c. 20
d. 45
e. none of the above.

4. The average case complexity analysis of an algorithm is
   a. the average of the best case complexity and the worst case complexity.
   b. the average number of operations for all inputs.
   c. the number of operations for input size $\frac{1}{n}(\sum_{i=1}^{n} i)$.
   d. half of the worst case complexity.
   e. none of the above.

5. The most efficient data structure in the worst case, among the following, in the cost of insertion is
   a. an open-addressing hash table.
   b. a binary search tree.
   c. a sorted array.
   d. an AVL tree.
   e. a stack.

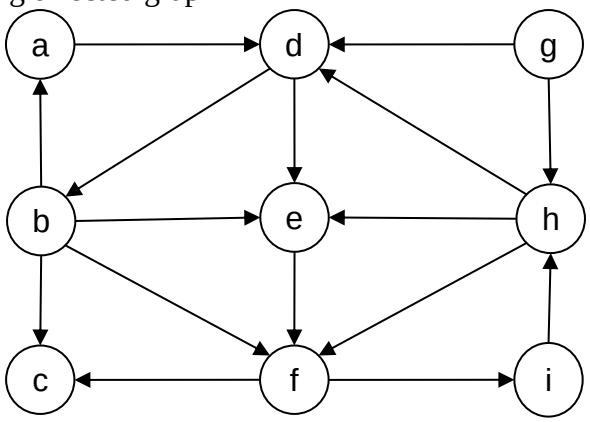
6. The method final in Question 2 when called on A=[5,4,3,2,1] and n = 2 outputs
   a. 4 3 4 5 5 5
   b. 5 4 5 3 5 4 5
   c. 0 0 0 1 0 0 0 3 0 0 0 1 0 0 0
   d. 5 5 5 4 5 5 5 3 5 5 5 4 5 5 5
   e. none of the above.

   5 5 5 4 5 5 5

7. The following statement is not true regarding recursive methods:
   a. Iterative methods cannot be converted to recursive methods.
   b. Non-tail recursive methods can be converted to tail-recursive methods.
   c. Non-tail recursive methods can be converted to iterative methods.
   d. A recursive method is excessively recursive if it repeats computations for some parameter values.
   e. A recursive method usually has less code than its corresponding iterative methods.

8. The following statement is not true regarding Huffman Coding:
   a. The Huffman Coding Tree is always a full binary tree.
   b. Huffman Coding is an example of static coding techniques requiring two passes.
   c. Huffman Coding is an example of a lossless compression technique.
   d. Some messages can be compressed using two distinct Huffman Coding Trees.
   e. In a given set of Huffman codewords, one codeword can be a prefix of another codeword.

9. Consider the following directed graph



The number of strongly connected components is equal to
   a. 1
   b. 2
   c. 3
   d. 4
   e. 5

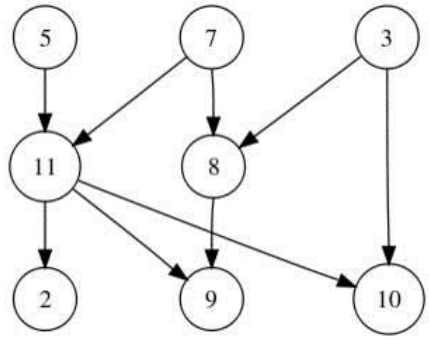
{g},{a,b,d,e,f,h,i},{c}

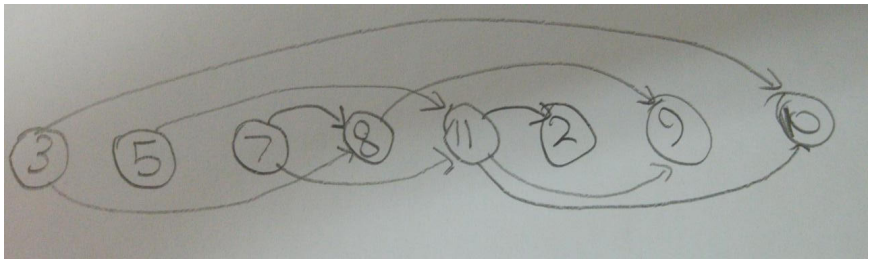
Q.2 [15 points] (Topological Sort):

a) [3 points] Is Topological sort applicable on any type of graph? If yes explain why. If no mention the properties required to apply Topological sort.

No, to apply topological sort, the graph should be directed, with no cycles. Also, not all (DAG) graphs can be sorted, there should always be a vertice with 0 in-degree at each step

b) [9 points] Show the steps of the topological sort algorithm on the following graph.

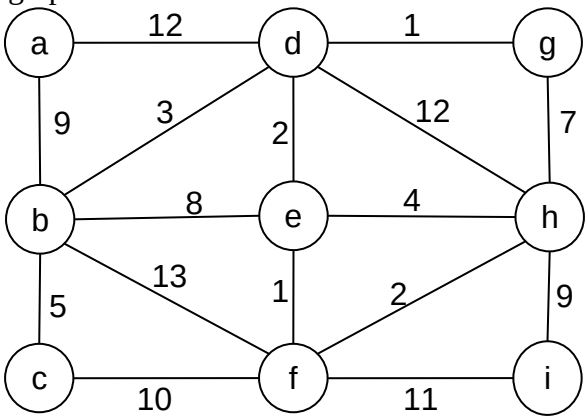




c) [3 points] Explain why the result of Topological sort may not be unique.

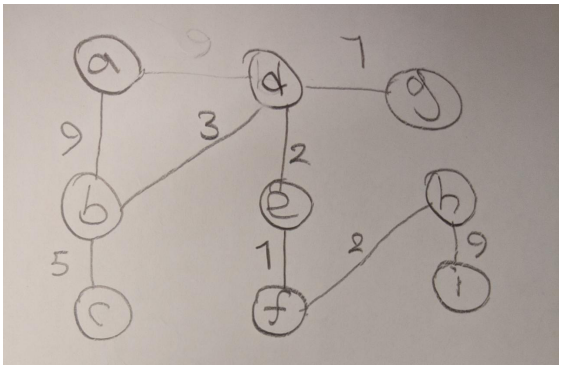
At a step, there might be multiple vertices with 0 in-degree, choosing any will make different toplogical graph, all correct

Q.3 [25 points] (Graphs):
Consider the following graph:



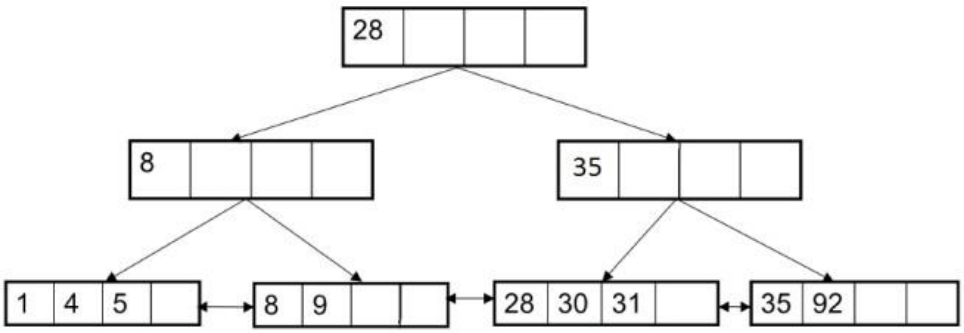
Pass	Initial distance	1	2	3	4	5	6	7	8	9	10	Edge Weight	Predecssor
Active vertex		b	d	g	e	f	h	c	i				
a	0	-	-	-	-	-	-	-	-	-	-	0	
b	9	-	-	-	-	-	-	-	-	-	-	9	a
c	-1	5	5	5	5	5	5	-	-	-	-	5	b
d	12	3	-	-	-	-	-	-	-	-	-	3	b
e	-1	8	2	2	-	-	-	-	-	-	-	2	d
f	-1	13	13	13	1	-	-	-	-	-	-	1	e
g	-1	-1	1	-	-	-	-	-	-	-	-	1	d
h	-1	-1	12	7	4	2	-	-	-	-	-	2	f
i	-1	-1	-1	-1	-1	11	9	9	-	-	-	9	h

a) (12 points) Carry out Prim's algorithm starting from vertex **a** to find the minimum spanning tree of the above graph, by filling the above table.
b) (5 points) Carry out Kruskal's algorithm on the above graph. Show all your intermediate work.
c) (4 points) Assuming that the visitor method prints the label of the vertex in the graph, show the output of the post-order depth first traversal of the above graph.
d) (4 points) Assuming that the visitor method prints the label of the vertex in the graph, show the output of the post-order depth first traversal of the above graph.



Same result for (a) and (b)
post-order: i,h,g,d,e,f,c,b,a
pre-order: a,b,c,f,e,d,g,h,i
*Alphabetic priority

Q.4 [20 points] (B+ Trees):
Consider the following B+ Tree:



a) [2 points] According to the above representation, what are the values of M and L for the B+ Tree?

L = 4
M = 5

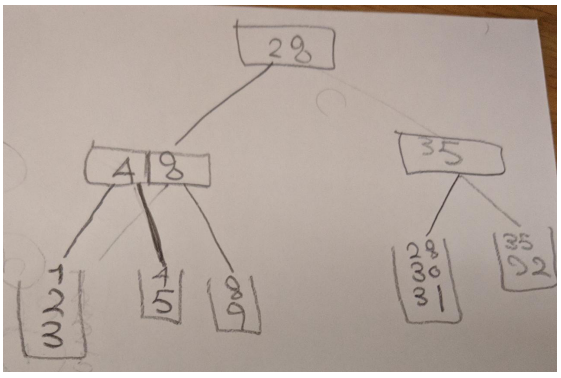
b) [3 points] When an internal node of the above B+ Tree becomes under-flow and when it becomes over-flow?

under: when keys<2 or subtree<3
over: when keys>4 or subtree>5

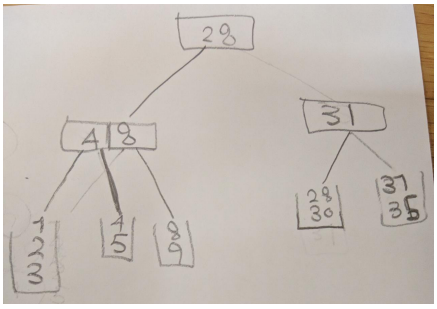
c) [3 points] When a leaf node of the above B+ Tree becomes under-flow and when it becomes over-flow?

under: when keys<2
over: when keys>4

d) [6 points] Show the above B+ Tree after inserting values 2 and 3.



e) [6 points] Show the above B+ Tree after deleting value 92.



Q.5 [20 points]: (Hashing and LZ-78 compression)

a) [10 points] Insert the following values in an open-addressing hash table of size 7 where the hashing function is K%13 and probing function is quadratic: $\pm i^2$:

14, 11, 17, 12, 27, 1, 40, 24, 53, 25

Mistake in question

b) [10 points] Compress the following message using LZ-78. (You must show the compression table):

BAABCAACCBAAABCA

code	index	word
(0,B)	1	B
(0,A)	2	A
(2,B)	3	AB
(0,C)	4	C
(2,A)	5	AA
(4,C)	6	CC
(1,A)	7	BA
(5,B)	8	AAB
(4,A)	9	CA