SET D

Name: Anvit Gupta Roll No. 22114009

Indian Institute of Technology Roorkee (IIT-R)

Spring Semester (2022-2023), End Term Examination

Sub: Data Structures (CSN/DA-102)

Class: B.Tech./B.Sc. I year

Time: 2 hours 30 minutes (2:30 PM to 05:00 PM)

Date: 12/06/2023

Max. Marks: 50

Instructions:

1. **Both Sections A and B are compulsory to attempt.**
2. **There is NO NEGATIVE marking for Section A.**
3. **In Section A, Questions 1 - 20 are multiple-choice questions. Each question has four options out of which one option is correct. Full marks will be awarded only if you mark the correct option.**

Section A

[Max. Marks: 30]

Q1. A stack is implemented with an array of 'A[0...M − 1]' and a variable 'tos'. The push and pop operations are defined by the following code.

```
push (x)
  A[tos] ← x
  tos ← tos − 1
end push

pop ( )
  tos ← tos + 1
  return A[tos]
end pop
```

Which of the following will initialize an empty stack with capacity M for the above implementation? **[1 Mark]**

a) tos ← 1
b) tos ← −1
c) tos ← M − 1
d) tos ← 0

Q2. What is the time complexity of $T(n) = \sqrt{n}\, T(\sqrt{n}) + n$? **[1 Mark]**

a) $O(n)$
b) $\theta(n \log \log n)$
c) $\Omega(n^2)$
d) $\theta(n \log n)$

Q3. Quicksort is run on two inputs shown below to sort in ascending order taking the first element as pivot,

(i) 1, 2, 3,......., n
(ii) n, n-1, n-2,......, 2, 1

Let C1 and C2 be the number of comparisons made for the inputs (i) and (ii) respectively. Then, **[2 Marks]**

a) We cannot say anything for arbitrary n
b) C1 < C2
c) C1 = C2
d) C1 > C2

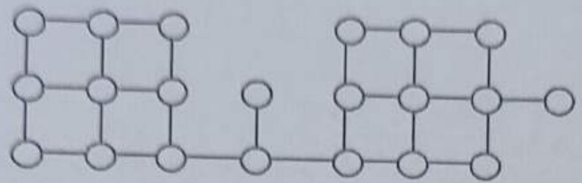
Q4. Let G be an undirected graph. Consider a depth-first traversal of G, and let T be the resulting depth-first search tree. Let u be a vertex in G and let v be the first new (unvisited) vertex visited after visiting u in the traversal. Which of the following statements is always true? **[2 Marks]**

a) If {u,v} is not an edge in G then u and v must have the same parent in T
b) {u,v} must be an edge in G, and u is a descendant of v in T

Name: Roll No.

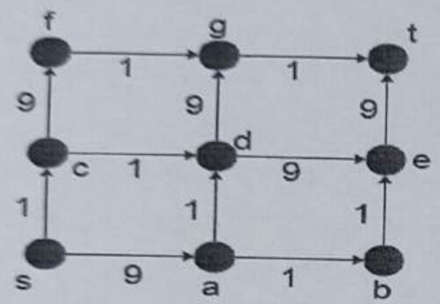
c) If {u,v} is not an edge in G then u is a leaf in T
d) {u,v} must be an edge in G, and v is a descendant of u in T

Q5. Suppose depth first search is executed on the graph below starting at some unknown vertex. Assume that a recursive call to visit a vertex is made only after first checking that the vertex has not been visited earlier. Then the maximum possible recursion depth (including the initial call) is ________. **[2 Marks]**



a) 20
b) 17
c) 19
d) 18

Q6. In a directed acyclic graph with a source vertex s, the quality-score of a directed path is defined to be the product of the weights of the edges on the path. Further, for a vertex v other than s, the quality-score of v is defined to be the maximum among the quality-scores of all the paths from s to v. The quality-score of s is assumed to be 1. The sum of quality-scores of all vertices on the graph shown below is ________ . **[2 Marks]**



a) 1023
b) 729
c) 81
d) 929

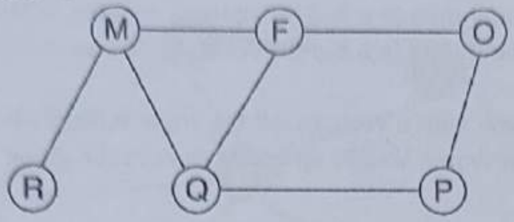
Q7. Let G=(V,E) be a directed, weighted graph with weight function w:E→R. For some function f:V→R, for each edge (u,v)∈E, define w′(u,v) as w(u,v)+f(u)−f(v). Which one of the options completes the following sentence so that it is TRUE ? "The shortest paths in G under w are shortest paths under w′ too, __________". **[1 Mark]**

a) if and only if f(u) is the distance from s to u in the graph obtained by adding a new vertex s to G and edges of zero weight from s to every vertex of G
b) for every f:V→R
c) if and only if ∀u∈V, f(u) is negative
d) if and only if ∀u∈V, f(u) is positive

Q8. What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search? **[1 Mark]**

a) $N*\log(N^2)$
b) N
c) N^2
d) $N*\log(N)$

Q9. The Breadth First Search algorithm has been implemented using the queue data structure. One possible order of visiting the nodes of the following graph is: **[1 Mark]**



a) QMFPOR
b) MFOPQR
c) QMFPRO
d) FQMPOR

Q10. Suppose there is a balanced binary search tree with n nodes, where at each node, in addition to the key, we store the number of elements in the subtree rooted at that node. Now, given two elements a and b, such that a<b , we want to find the number of elements x in the tree that lie between a and b, that is, $a \leq x \leq b$. This can be done with (choose the best solution). **[2 Marks]**

a) O(n) comparisons and O(n) additions, using depth-first- search
b) O(log n) comparisons and O(log n) additions
c) O(log n) comparisons but no further additions
d) O(log n) comparisons but a constant number of additions
e) O($\sqrt{n}$) comparisons but O(log n) additions

Q11. The first n cells of an array L contain positive integers sorted in decreasing order, and the remaining m − n cells all contain 0. Then, given an integer x, in how many comparisons can one find the position of x in L? **[1 Mark]**

a) O(log(m/n)) comparisons suffice
b) At least n comparisons are necessary in the worst case
c) At least log m comparisons are necessary in the worst case
d) O(log n) comparisons suffice
e) O(log(m − n)) comparisons suffice

Q12. Two matrices M1 and M2 are to be stored in arrays A and B respectively. Each array can be stored either in row-major or column-major order in contiguous memory locations. The time complexity of an algorithm to compute M1 × M2 will be: **[1 Mark]**

a) independent of the storage scheme
b) best if A is in row-major, and B is in column- major order
c) best if both are in column-major order
d) best if both are in row-major order

Q13. Consider the following code fragment in the C programming language when run on a non-negative integer n.

```
int f(int n)
{
if(n==0 || n==1)
return 1;
else
return f(n-1) + f(n-2);
}
```

Assuming a typical implementation of the language, what is the running time of this algorithm and how does it compare to the optimal running time for this problem? **[2 Marks]**

a) The algorithm does not terminate

Name: Roll No.

b) This algorithm runs in polynomial time in n but the optimal running time is exponential in n
c) This algorithm runs in exponential time in n and the optimal running time is exponential in n
d) This algorithm runs in polynomial time in n and the optimal running time is polynomial in n
e) This algorithm runs in exponential time in n but the optimal running time is polynomial in n

Q14. In a connected weighted graph with n vertices, all the edges have distinct positive integer weights. Then, the maximum number of minimum weight spanning trees in the graph is ________. **[1 Mark]**

a) n^{n-2}
b) 1
c) n
d) equal to maximum weight of an edge of the graph
e) equal to number of edges in the graph

Q15. A binary search tree contains the numbers 1, 2, 3, 4, 5, 6, 7, 8. When the tree is traversed in pre-order and the values in each node printed out, the sequence of values obtained is 5, 3, 1, 2, 4, 6, 8, 7. If the tree is traversed in post-order, the sequence obtained would be: **[2 Marks]**

a) 2, 1, 4, 3, 7, 8, 6, 5
b) 8, 7, 6, 5, 4, 3, 2, 1
c) 2, 1, 4, 3, 6, 7, 8, 5
d) 1, 2, 3, 4, 8, 7, 6, 5

Q16. A hash table of length 10 uses open addressing with hash function h(k)=k mod 10, and linear probing. After inserting 6 values into an empty hash table, the table is as shown below. Which one of the following choices gives a possible order in which the key values could have been inserted in the table? **[2 Marks]**

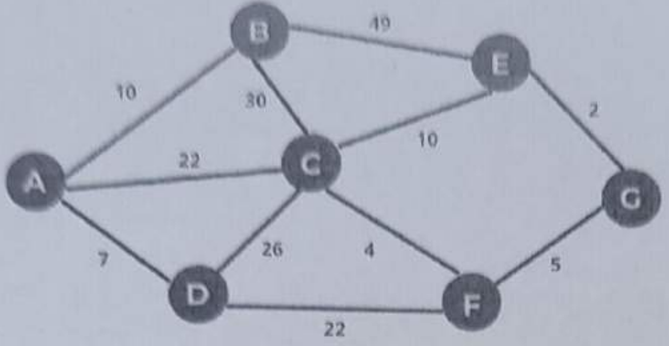
0	
1	
2	42
3	23
4	34
5	52
6	46
7	33
8	
9	

a) 42, 46, 33, 23, 34, 52
b) 46, 42, 34, 52, 23, 33
c) 46, 34, 42, 23, 52, 33
d) 34, 42, 23, 52, 33, 46

Q17. For an undirected graph G = (V,E), the line graph G = (V', E') is obtained by replacing each edge in E by a vertex, and adding an edge between two vertices in V' if the corresponding edges in G are incident on the same vertex. Which of the following is TRUE of line graphs? **[2 Marks]**

a) each vertex in the line graph has degree one or two
b) the line graph for a complete graph is complete
c) the line graph for a connected graph is connected
d) the maximum degree of any vertex in the line graph is at most the maximum degree in the original graph
e) the line graph for a bipartite graph is bipartite

Q18. Consider the undirected graph below:



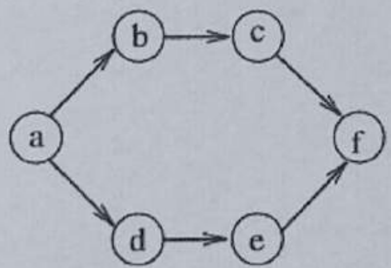
Using Prim's algorithm to construct a minimum spanning tree starting with node A, which one of the following sequences of edges represents a possible order in which the edges would be added to construct the minimum spanning tree? **[2 Marks]**

a) (A, D), (A, B), (D, F), (F, C), (F, G), (G, E)
b) (E, G), (C, F), (F, G), (A, D), (A, B), (A, C)
c) (A, B), (A, D), (D, F), (F, G), (G, E), (F, C)
d) (A, D), (A, B), (A, C), (C, F), (G, E), (F, G)

Q19. Let G be an undirected connected graph with distinct edge weights. Let e_max be the edge with maximum weight and e_min be the edge with minimum weight. Which of the following statements is false? **[1 Mark]**

a) Every minimum spanning tree of G must contain e_min
b) G has a unique minimum spanning tree
c) If e_max is in a minimum spanning tree, then its removal must disconnect G
d) No minimum spanning tree contains e_max

Q20. Consider the following directed acyclic graph "G" with vertex set V={a,b,c,d,e,f}:



Which of the following is not a topological ordering of G? **[1 Mark]**

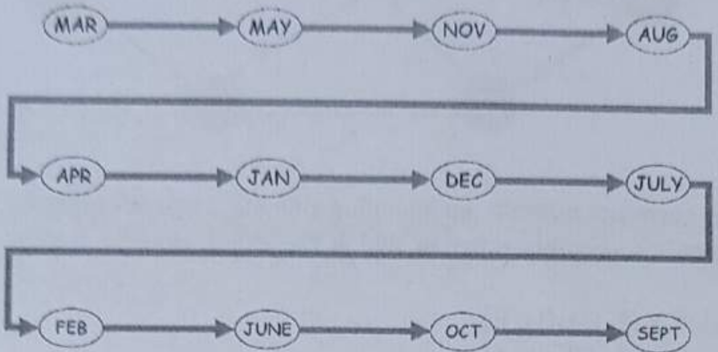
a) a d b c e f
b) a b c d e f
c) a b d c e f
d) a d e b c f

Name: Roll No. SET D

Section B

[Max. Marks: 20]

Q1. Using the following ordered sequence of month names, construct an AVL-tree by following AVL insertion rules. Demonstrate all steps and rotations during the tree's construction. **[5 Marks]**



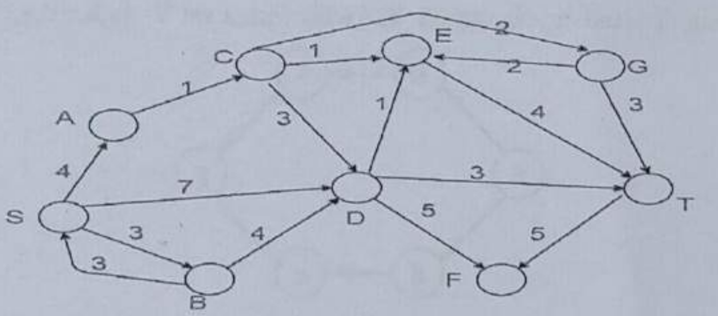
Q2. Consider the directed graph shown in the figure below. There are multiple shortest paths between vertices S and T. Verify all the paths through stepwise demonstration using Dijkstra's shortest path algorithm, and based on the outcomes determine the paths belongs to Dijkstra's shortest path (Yes/No). **[2+2+2 Marks]**

(a) SBDT

(b) SACDT

(c) SACET

[Hint: Assume that, in any iteration, the shortest path to a vertex v is updated only when a strictly shorter path to v is discovered.]



Q.3. Create a B-tree t=3 [illegible]-way using these:

3, 7, 9, 23, 45, 1, 5, 14, 25, 24, 13, 11, 8, 19, 4, 31, 35, 56

Insert these further keys: 2, 6,12

Delete these keys: 4, 5, 7, 3, 14

Demonstrate all steps for B-tree creation, and key insertions and deletions. **[3+2+2 Marks]**

Q.4. Consider a double hashing scheme in which the primary hash function is $h1(k) = k \bmod 23$ and the secondary hash function is $h2(k) = 1 + (k \bmod 19)$. Assume that the table size is 23. Then, what is the address returned by probe 1 in the probe sequence (assume that the probe sequence begins at probe 0) for key value $k = 90$? **[2 Marks]**

Good Luck!