



# CS502 Fundamentals Of Algorithms Update MCQS For Quiz-3 File Solve By Vu Topper RM



**80 To 100% Marks**



وَتَعَزُّ مِنْ تَشَاءِ وَتُدُلُّ مِنْ تَشَاءِ



## PROFESSIONAL ONLINE ACADEMY

### WE Offers

LMS Handling

Important Notes

Online Classes

Assignments

Quiz & GDB's

Projects

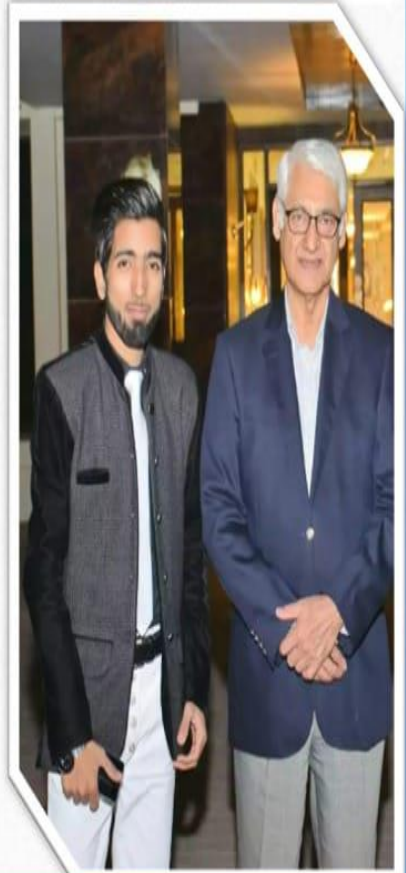
**NOTHING  
IS  
IMPOSSIBLE**

**Join Us  
Now**

For More Info  
Contact us at:

**Rizwan Manzoor**

☎ **0322-4021365**



بری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

Equivalence relation partitions the vertices into \_\_\_\_\_ classes of mutually reachable vertices and these are the strong components

- A. Variance
- B. Non classes
- C. Equivalence**
- D. Non equivalence

**Page 136**

The ancestor and descendent relation can be nicely inferred by the \_\_\_\_\_ lemma.

- A. Node
- B. Division
- C. Addition
- D. Parenthesis**

**Page 129**

Networks are \_\_\_\_\_ in the sense that it is possible from any location in the network to reach any other location in the digraph.

- A. Not graphs.
- B. Complete**
- C. Incomplete
- D. Transportation

**Page 135**

In strong components algorithm, first of all DFS is run for getting \_\_\_\_\_ times of vertices.

- A. Start**
- B. Finish
- C. Middle
- D. Both start & finish

**Page 138**

Timestamp structure of \_\_\_\_\_ is used in determining the strong components of a digraph.

- A. DFS**
- B. BFS
- C. MST
- D. Both DFS, BFS

**Google**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

In Timestamped DFS-cycles lemma, if edge  $(u, v)$  is a tree, forward or cross edge, then \_\_\_\_\_

**A.  $f[u] > f[v]$**

B.  $f[u] < f[v]$

C.  $f[u] \leq f[v]$

D.  $f[u] \geq f[v]$

The component digraph is necessarily \_\_\_\_\_.

A. Cyclic

B. Strong

**C. Acyclic**

**Page 136**

D. Straight

In computing the \_\_\_\_\_ components of a digraph, vertices of the digraph are partitioned into subsets.

A. Best

B. Worst

C. Weakly connected

**D. Strongly connected**

**Page 135**

There exist a unique path between any \_\_\_\_\_ vertices of a free tree.

A. One

**B. Two**

C. five

D. three

The \_\_\_\_\_ given by DFS allow us to determine whether the graph contains any cycles.

A. Order

**B. Time stamps**

**Page 130**

C. BFS traversing

D. Topological sort

Forward edge is:

A.  $(u, v)$  where  $u$  is a proper descendent of  $v$  in the tree.

**B.  $(u, v)$  where  $v$  is a proper descendent of  $u$  in the tree.**

**Page 129**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- C. (u, v) where v is a proper ancestor of u in the tree.
- D. (u, v) where u is a proper ancestor of v in the tree.

Once you enter a strong component, every vertex in the component is \_\_\_\_\_.

- A. Removed
- B. Reachable** **Page 137**
- C. not reachable
- D. reachable some times

In Generic approach determining of Greedy MST, we maintain a subset A of \_\_\_\_\_ .

- A. Paths
- B. Edges** **Page 143**
- C. Cycles
- D. Vertices

If you find yourself in maze the better traversal approach will be :

- A. DFS
- B. BFS**
- C. Level order
- D. DFS

Computing the strongly connected components of a digraph is a/an \_\_\_\_\_ of the problem to determine whether a digraph is strongly connected or not.

- A. Size
- B. Connection
- C. Optimization
- D. Generalization** **Page 135**

A free tree with n \_\_\_\_\_ have exactly n-1 \_\_\_\_\_.

- A. vertices, edges** **Page 142**
- B. vertices, nodes
- C. nodes, vertices
- D. vertices, nodes

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

\_\_\_\_\_ components are not affected by reversal of all edges in terms of vertices reachability.

- A. First two
- B. Last two
- C. Weakly connected
- D. Strongly connected**

**Page 139**

Adding any edge to a free tree creates a unique \_\_\_\_\_ .

- A. Vertex
- B. Cycle**
- C. Edge
- D. Strong component

**Page 142**

In computing the strongly connected components of a digraph, vertices of the digraph are \_\_\_\_\_ into subsets.

- A. Joined
- B. Deleted
- C. Created
- D. Partitioned**

**Page 135**

For \_\_\_\_\_ graphs, there is no distinction between forward and back edges.

- A. Large
- B. Medium
- C. Directed
- D. Undirected**

**Page 130**

A topological sort of a DAG is a \_\_\_\_\_ ordering of the vertices of the DAG such that for each edge  $(u, v)$ ,  $u$  appears before  $v$  in the ordering.

- A. Linear**
- B. Parallel
- C. Sequence
- D. Non-Linear

**Page 134**

By breaking any edge on a cycle created in free tree, the free \_\_\_\_\_ is restored.

- A. Tree**

**Page 142**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- B. Edge
- C. Cycle
- D. Vertex

Which technique is used in the implementation of Kruskal solution for the MST?

- A. Greedy Technique** **Page 142**
- B. Divide-and-Conquer Technique
- C. Dynamic Programming Technique
- D. The algorithm combines more than one of the above techniques i.e. Divide-and-Conquer and Dynamic Programming

A strongly connected component only apply to:

- A. Directed Graph** **Page 135**
- B. Undirected Graph
- C. Breadth First Search
- D. Minimum Spanning Tree

The relationship between number of back edges and number of cycles in DFS is,

- A. Both are equal
- B. Back edges are half of cycles
- C. Back edges are one quarter of cycles
- D. There is no relationship between no. of edges and cycles**  
**Page 131**

Digraphs \_\_\_\_\_ in communication and transportation networks.

- A. are used** **Page 135**
- B. are not used
- C. parts are used
- D. final value is used

Back edge is:

- A. (u, v) where v is an ancestor of u in the tree.** **Page 128**
- B. (u,v) where u is an ancestor of v in the tree.
- C. (u, v) where v is a predecessor of u in the tree.
- D. None of above

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

There are no \_\_\_\_\_ edges in undirected graph.

- A. Back
- B. Both
- C. Cross**
- D. Forward

**Page 130**

In strong components algorithm, the form of graph is used in which all the \_\_\_\_\_ of original graph G have been reversed in direction.

- A. Edges**
- B. Trees
- C. Vertices
- D. Both edges & vertices

**Page 138**

Which activity creates a unique cycle in a free tree:

- A. adding root
- B. adding any edge**
- C. adding any vertex
- D. adding any sub tree

**Page 142**

You have an adjacency list for G, what is the time complexity to compute Graph transpose  $G^T$ ?

- A.  $(V + E)$**
- B.  $(V E)$
- C.  $(V)$
- D.  $(V^2)$

**Page 138**

The time complexity to compute Graph transpose  $G^T$  is  $(V+E)$ , if you have \_\_\_\_\_ for G.

- A. Stack
- B. Array list
- C. Complete list

**D. An adjacency list** **Page 138**

In undirected graph, by convention all the edges are called \_\_\_\_\_ edges.

- A. Back**
- B. Cross

**Page 130**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- C. Forward
- D. Both forward and back

The \_\_\_\_\_ given by DFS allow us to determine a number of things about a graph or digraph.

- A. line stamps
- B. node stamps
- C. color stamps
- D. time stamps**

**Page 130**

If a subset of edges A is viable for building MST, it can not contain a/an \_\_\_\_\_.

- A. Edge
- B. Cycle**
- C. Graph
- D. Vertex

**Page 143**

A free tree with n vertices have exactly \_\_\_\_\_ edges.

- A. 1
- B. N
- C. n-1**
- D. n+1

**Page 142**

\_\_\_\_\_ technique is look like propagating wave-front outward.

- A. Generic Traversal
- B. Depth First Traversal
- C. Time Stamp Traversal
- D. Breath first traversal**

**Page 117**

In Timestamped DFS, If there is a back edge (u, v) then v is an ancestor of u and by following tree edge from v to u, we get \_\_\_\_\_.

- A. a line
- B. a cycle**
- C. a graph
- D. nothing

**Page 131**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

In strong components algorithm, vertices are sorted in \_\_\_\_\_ order of finish times.

- A. Any
- B. Strong
- C. Increasing
- D. Decreasing**

**Page 141**

A fully connected undirected graph of 5 nodes will have \_\_\_\_\_ edges.

- A. 4
- B. 5
- C. 10**
- D. 15

A digraph is strongly connected under what condition?

- A. A digraph is strongly connected if for every pair of vertices  $u, v \in V$ ,  $u$  can reach  $v$ .
- B. A digraph is strongly connected if for every pair of vertices  $u, v \in V$ ,  $u$  can reach  $v$  and vice versa.**
- C. A digraph is strongly connected if for at least one pair of vertex  $u, v \in V$ ,  $u$  can reach  $v$  and vice versa.
- D. A digraph is strongly connected if at least one third pair of vertices  $u, v \in V$ ,  $u$  can reach  $v$  and vice versa.

We say that two vertices  $u$  and  $v$  are mutually \_\_\_\_\_ if  $u$  can reach  $v$  and vice versa.

- A. Crossed
- B. Forward
- C. Reachable**
- D. Not Reachable

**Page 135**

In Timestamped DFS, No back edges means \_\_\_\_\_.

- A. DFS
- B. BFS
- C. 1 cycle
- D. no cycles**

**Page 131**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

Cross edge is :

- A. (u, v) where u and v are not ancestor of one another
- B. (u, v) where u is ancestor of v and v is not descendent of u.
- C. (u, v) where u and v are not ancestor or descendent of one another**
- D. (u, v) where u and v are either ancestor or descendent of one another.

In digraph  $G=(V,E)$  ;G has cycle if and only if

- A. The DFS forest has forward edge.
- B. The DFS forest has back edge**
- C. The DFS forest has both back and forward edge
- D. BFS forest has forward edge

**Page 131**

**Question No:01**

**(Marks:01)**

**Vu-Topper RM**

If we encode and compress text using ASCII standard each character is represented b

- A. Fixed length code word of 4 bits
- B. Variable length code word up to 4 bits
- C. Fixed length code word of 8 bits**
- D. Variable length code word up to 8 bits

**Question No:02**

**(Marks:01)**

**Vu-Topper RM**

G power T for graph can be computed in

- A.  $\Theta(V E)$
- B.  $\Theta(V \log E)$
- C.  $\Theta(E \log V)$
- D.  $\Theta(V+E)$**

**Question No:03**

**(Marks:01)**

**Vu-Topper RM**

Adding any edge to free tree

- A. Keep it free tree and increase the size of the tree**
- B. Create a unique cycle
- C. Not allow to add edge to free tree
- D. Create multiple cycles

**Question No:04**

**(Marks:01)**

**Vu-Topper RM**

Which sorting algorithm is faster :

- A.  $O(n^2)$
- B.  $O(n \log n)$

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

C.  $O(n+k)$

D.  $O(n^3)$

**Question No:05**

**(Marks:01)**

**Vu-Topper RM**

**Question No:06**

**(Marks:01)**

**Vu-Topper RM**

**Question No:07**

**(Marks:01)**

**Vu-Topper RM**

Networks are complete in the sense that it is possible from any location in the network to reach any other location in the digraph.

**A. True**

B. False

**Question No:08**

**(Marks:01)**

**Vu-Topper RM**

Runtime complexity of Prim's algorithm is \_\_\_\_\_.

A.  $V \log V$

**B.  $E \log V$**

C.  $\log V$

D. None of the above

**Question No:09**

**(Marks:01)**

**Vu-Topper RM**

In Prim's algorithm, we start with the \_\_\_\_\_ vertex  $r$ ; it can be any vertex.

A. First

B. Leaf

**C. root**

D. Mid

**Question No:10**

**(Marks:01)**

**Vu-Topper RM**

Adding any edge to a free tree creates a unique cycle.

**A. true**

B. false

**Question No:11**

**(Marks:01)**

**Vu-Topper RM**

For undirected graph, there is no distinction between forward and back edges.

**A. true**

B. false

**Question No:12**

**(Marks:01)**

**Vu-Topper RM**

in strong components algorithm, first of all DFS is run for computing finish times of vertices.

**A. true**

B. false

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:13**

**(Marks:01)**

**Vu-Topper RM**

Which method is preferable for dealing with chain matrix multiplication?

- A. Divide and conquer strategy
- B. Dynamic programming formulation**
- C. Graph theory
- D. Greedy Approach

**Question No:14**

**(Marks:01)**

**Vu-Topper RM**

Huffman algorithm produces the.....prefix code tree.

- A. Better
- B. Optimal**
- C. Worst
- D. Best

**Question No:15**

**(Marks:01)**

**Vu-Topper RM**

A...w is adjacent to vertex v if there is an edge from v to w.

- A. Acyclic
- B. Vertex**
- C. Loop
- D. Cycle

**Question No:16**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string "greedy" will be encoded with

- A. 44 bits
- B. 120 bits
- C. 40 bits
- D. 48 bits**

**Question No:17**

**(Marks:01)**

**Vu-Topper RM**

In activity scheduling algorithm, each activity is represented by a

- A. Rectangle**
- B. Square
- C. Circle
- D. Triangle

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:18**

**(Marks:01)**

**Vu-Topper RM**

Those problems in which greedy finds good, but not always best is called a greedy.....

**A. Heuristic**

B. Solution

C. Result

D. Algorithm

**Question No:19**

**(Marks:01)**

**Vu-Topper RM**

The knapsack problem belongs to the domain of.....Problems

A. Searching

B. Sorting

C. Linear solution

**D. Optimization**

**Question No:20**

**(Marks:01)**

**Vu-Topper RM**

The general coin change problem can be solved using

A. Recursion

B. Greedy algorithm

**C. Dynamic programming**

D. Divide and conquer

**Question No:21**

**(Marks:01)**

**Vu-Topper RM**

Huffman algorithm generates an optimum.....code

A. Postfix

B. Infix

C. None of the given options

**D. Prefix**

**Question No:22**

**(Marks:01)**

**Vu-Topper RM**

.....ways of representing graphics

A. 1

**B. 2**

C. 3

D. 4

**Question No:23**

**(Marks:01)**

**Vu-Topper RM**

Knapsack word originates from.....language

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**A. German**

B. English

C. French

D. Norwegian

**Question No:24**

**(Marks:01)**

**Vu-Topper RM**

Graphs are important ..... model for many application problems

**A. Mathematical**

B. Unpredictable

C. Haphazard

D. Unsystematic

**Question No:25**

**(Marks:01)**

**Vu-Topper RM**

Which type of algorithm is harder to prove the correctness?

A. Dynamic

**B. Greedy**

C. Divide and conquer

D. Brute force

**Question No:26**

**(Marks:01)**

**Vu-Topper RM**

Items are not allowed in 0/1 knapsack problem

**A. Fractional**

B. 0

C. 1

D. 0/1

**Question No:27**

**(Marks:01)**

**Vu-Topper RM**

Matrix multiplication is a(n)..... operation

A. Neither commutative nor associative

B. Transitive

C. Commutative

**D. Associative**

**Question No:28**

**(Marks:01)**

**Vu-Topper RM**

For a Diagraph  $G = (V, E)$ , Sum of in-degree (v) ---.

A. Not equal to sum of out-degree(v)

**B. = sum of out-degree(v)**

C. < sum of out-degree(v)

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

D.  $>$  sum of out-degree(v)

**Question No:29** (Marks:01)

**Vu-Topper RM**

DFS or BFS yields a \_\_\_\_\_ of the graph.

A. Traversed tree

**B. Spanning tree** Page 125

C. Simple tree

D. Free tree

**Question No:30** (Marks:01)

**Vu-Topper RM**

Using ASCII code, each character is represented by a fixed-length code of bits per character.

A. 4

B. 6

C. 10

**D. 8**

**Question No:31** (Marks:01)

**Vu-Topper RM**

In Knapsack Problem, the goal is to put items in the Knapsack such that the value of the items is-----subject to weight limit of the Knapsack.

A. Minimized

B. Decreased

**C. Maximized**

D. None of the above given

**Question No:32** (Marks:01)

**Vu-Topper RM**

A graph is said to be acyclic if it contains ---.

A. At least one cycle

B. Exactly one cycle

C. Always more than one cycle

**D. No cycles**

**Question No:33** (Marks:01)

**Vu-Topper RM**

The number of edges that come out of a vertex is called the \_\_\_\_\_ of that vertex in the digraph.

A. Post-degree

B. in-degree

**C. out-degree**

D. pre-degree

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:34** (Marks:01) **Vu-Topper RM**

If Matrix-A has dimensions "3x2" and Matrix-B has dimensions "2x3", then multiplication of Matrix-A and Matrix-B will result a new Matrix-C having dimensions.

- A. 3x2
- B. 2x3
- C. 2x2
- D. 3x3**

**Question No:35** (Marks:01) **Vu-Topper RM**

A/an is one in which you want to find, not just a solution, but the best solution.

- A. Optimization problem**
- B. Divide and Conquer
- C. NP complete problem
- D. Best problem

**Question No:36** (Marks:01) **Vu-Topper RM**

Fractional Knapsack is founded on -----method.

- A. Greedy page**
- B. Recursive
- C. Divide and Conquer
- D. Dynamic programming

**Question No:37** (Marks:01) **Vu-Topper RM**

Find the maximum value of the items which can carry using knapsack weight capacity

=50

ITEM	10	20	30	70
WEIGHT				
VALUE	70	20	80	200

- A. 90
- B. 280
- C. 200
- D. 100**

**Question No:38** (Marks:01) **Vu-Topper RM**

If the graph is represented using an adjacency matrix, then Breadth-first search takes-----time.

- A.  $O(E+1)$
- B.  $O(V^2)$**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- C. O(V)
- D. O(E)

**Question No:39** (Marks:01) **Vu-Topper RM**

In inductive approach of Knapsack problem, we consider 2 cases, -----  
or --.Median, Mode

- A. Recursive, Iterative
- B. Leave object, Take object**
- C. Sequentially, Parallel

**Question No:40** (Marks:01) **Vu-Topper RM**

A Greedy algorithm can NOT be used to solve all the problems.

- A. Dynamic programming**
- B. Memorization programming
- C. Edit-distance programming
- D. Storing value programming

**Question No:41** (Marks:01) **Vu-Topper RM**

In Huffman encoding, the \_\_\_\_\_ is the number of occurrences of a character divided by the total characters in the message.

- A. Counting
- B. Parsing
- C. Relative Probability**
- D. Weight

**Question No:42** (Marks:01) **Vu-Topper RM**

The Binary Tree constructed by a Huffman Encoding is a:

- A. Full Binary Tree**
- B. Partial Binary Tree
- C. Incomplete Binary Tree
- D. None of the given option

**Question No:43** (Marks:01) **Vu-Topper RM**

Following is not the application of Edit Distance Problem.

- A. Speech recognition**
- B. Spelling correction
- C. Ascending order
- D. Computational Molecular Biology

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:44**

**(Marks:01)**

**Vu-Topper RM**

Consider three Matrices X, Y, Z of dimensions  $1 \times 2$ ,  $2 \times 3$ ,  $3 \times 4$  respectively.  
The number of multiplication of (XYZ) is:

**A. 18**

B. 32

C. 24

D. 30

**Question No:45**

**(Marks:01)**

**Vu-Topper RM**

In----Knapsack Problem, limitation is that an item can either be put in the bag or not. Fractional items are not allowed.

A. 0

B. 1

**C. 0/1**

D. Fractional

**Question No:46**

**(Marks:01)**

**Vu-Topper RM**

An in-place sorting algorithm is one that uses additional array for storage.

A. Always

B. Permanently

**C. Does not**

D. Sometime

**Question No:47**

**(Marks:01)**

**Vu-Topper RM**

If Matrix-A has dimensions " $p \times q$ " and Matrix-B has dimensions " $q \times r$ ", then multiplication of Matrix-A and Matrix-B will result a new Matrix-C having dimensions.

A.  $P \times q$

**B.  $P \times r$**

C.  $q \times r$

D.  $q \times p$

**Question No:48**

**(Marks:01)**

**Vu-Topper RM**

Counting sort is suitable to sort the elements in range 1 to K.

A. K is large

**B. K is small**

C. K may be large or small

D. None

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:49**

**(Marks:01)**

**Vu-Topper RM**

When matrix A of  $5 \times 3$  is multiply with matrix B of  $3 \times 4$  then the multiplication required is:

- A. 15
- B. 12
- C. 36
- D. 60**

**Question No:50**

**(Marks:01)**

**Vu-Topper RM**

----- is a linear time sorting algorithm.

- Merge sort
- Quick sort
- Bubble sort
- Radix sort**

**Question No:51**

**(Marks:01)**

**Vu-Topper RM**

In Dynamic Programming approach, we do not store the solution to each sub problem in case if it reappears.

- A. True
- B. False**

**Question No:52**

**(Marks:01)**

**Vu-Topper RM**

Dynamic Programming approach is usually useful in solving optimization problem.

- A. True**
- B. False

**Question No:53**

**(Marks:01)**

**Vu-Topper RM**

Which of the following algorithm provides an optimal solution for the activity selection problem?

- A. Divide and Conquer
- B. Brute force
- C. Greedy**
- D. Recursive

**Question No:54**

**(Marks:01)**

**Vu-Topper RM**

A graph is if every vertex can reach every other vertex.

- A. Connected**
- B. Cycle

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- C. Acyclic
- D. Loop

**Question No:55** (Marks:01) **Vu-Topper RM**

In a Huffman encoding when a new node is created by combining two nodes, the new node is placed in the \_\_.

- A. Priority queue**
- B. Linked list
- C. Min heap tree
- D. Graph traversal

**Question No:56** (Marks:01) **Vu-Topper RM**

In \_\_ algorithm, you hope that by choosing a local optimum at each step, you will end up a global optimum.

- A. Simple
- B. Divide and conquer
- C. Greedy**
- D. Brute Force

**Question No:57** (Marks:01) **Vu-Topper RM**

The string "Imncde" is coded with ASCII code, the message length would be \_\_ bits.

- A. 24
- B. 36
- C. 48**
- D. 60

**Question No:58** (Marks:01) **Vu-Topper RM**

For graph traversal, breadth-first search strategy \_\_\_\_\_

- A. Is always recursive
- B. Cannot be recursive
- C. Cannot be non-recursive
- D. Can be both recursive and non-recursive**

**Question No:59** (Marks:01) **Vu-Topper RM**

In activity scheduling algorithm, the width of a rectangle \_\_\_\_\_

- A. Is always ignored
- B. Directs towards recursion
- C. Should be maximized
- D. Indicates the duration of an activity**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:60**

**(Marks:01)**

**Vu-Topper RM**

If the graph is represented using an adjacency list, then Breadth-first search take time

- A.  $O(V^2)$
- B.  $O(V)$
- C.  $O(V+E)$**
- D.  $O(E+1)$

**Question No:61**

**(Marks:01)**

**Vu-Topper RM**

Suppose you are given infinite coins of 1,2 ,3, and 4.Select the ways of the minimum numberof coins that required to achieve a sum of 6:

- A. 1
- B. 2**
- C. 3
- D. 4

**Question No:62**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string “greedy” will be encoded with

- A. 48 bitS**
- B. 20 bits
- C. 44 bits
- D.40 bits

**Question No:63**

**(Marks:01)**

**Vu-Topper RM**

The Huffman codes provide a method of data efficiency.

- A. Reading/Writing
- B.Encoding/Decoding**
- B. Divide/Conquer
- C. Inserting/Deleting

**Question No:64**

**(Marks:01)**

**Vu-Topper RM**

In the context of activity selection algorithm, time s dominated by sorting of the activities by-----.

- A. Start Times
- B. Finish Times**
- C. Average Times
- D.CPU Burst Times

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:65** (Marks:01) **Vu-Topper RM**

Time complexity of the “0-1” knapsack algorithm depends on----

- A. Number of items
- B. Capacity of the knapsack
- C. Size of the Table
- D. Number of items and capacity of knapsack**

**Question No:66** (Marks:01) **Vu-Topper RM**

The greedy approach gives us an optimal solution when the coins are all powers of a -----denomination

- A. Fixed**
- B. Variable
- C. Constant
- D. Static

**Question No:67** (Marks:01) **Vu-Topper RM**

In Activity Selection, we say that two activities are non-interfering if their start-finish interval overlap

- A. Do
- B. Do not**
- C. Sometimes
- D. Once

**Question No:68** (Marks:01) **Vu-Topper RM**

How many steps are involved to design the dynamic programming strategy?

- A. 2
- B. 3
- C. 1
- D. 4**

**Question No:69** (Marks:01) **Vu-Topper RM**

Bag is a.....

- A. type of algorithm**
- B. data structure
- C. program
- D. compiler

**Question No:70** (Marks:01) **Vu-Topper RM**

If a problem is in NP-complete, it must also be in NP.

- A. True**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

B.False

**Question No:71 (Marks:01)**

**Vu-Topper RM**

The Huffman algorithm finds a optimal solution.

**A. True**

B. false

**Question No:72 (Marks:01)**

**Vu-Topper RM**

The Huffman algorithm finds an exponential solution

A. True

**B. False**

**Question No:73 (Marks:01)**

**Vu-Topper RM**

The Huffman algorithm finds a polynomial solution

**A. True**

B. False

**Question No:74 (Marks:01)**

**Vu-Topper RM**

The greedy part of the Huffman encoding algorithm is to first find two nodes withs sallest frequency.

**A. True**

B. False

**Question No:75 (Marks:01)**

**Vu-Topper RM**

The code word assigned to characters by the Huffman algorithm have the property that nocode word is the prefix of any other.

**A. True**

B.False

**Question No:76 (Marks:01)**

**Vu-Topper RM**

Dijkestra's single source shortest path algorithm works if all edges weights are non-negativeand there are negative cost cycles.

A. True

**B. False**

**Question No:77 (Marks:01)**

**Vu-Topper RM**

The term "coloring" came form the original application which was in architectural design.

A. True

**B. False**

**Question No:78 (Marks:01)**

**Vu-Topper RM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

In the clique cover problem, for two vertices to be in the same group, they must be adjacent to each other.

**A. True**

B. False

**Question No:79**

**(Marks:01)**

**Vu-Topper RM**

Dijkstra's algorithm is operated by maintaining a subset of vertices

**A. True**

B. False

**Question No:80**

**(Marks:01)**

**Vu-Topper RM**

We do sorting to,

A. keep elements in random positions

B. keep the algorithm run in linear order

C. keep the algorithm run in  $(\log n)$  order

**D. keep elements in increasing or decreasing order**

**Question No:81**

**(Marks:01)**

**Vu-Topper RM**

After partitioning array in Quick sort, pivot is placed in a position such that

**A. Values smaller than pivot are on left and larger than pivot are on right**

B. Values larger than pivot are on left and smaller than pivot are on right

C. Pivot is the first element of array

D. Pivot is the last element of array

**Question No:82**

**(Marks:01)**

**Vu-Topper RM**

Merge sort is stable sort, but not an in-place algorithm

**A. True**

B. False

**Question No:83**

**(Marks:01)**

**Vu-Topper RM**

A  $p \times q$  matrix A can be multiplied with a  $q \times r$  matrix B. The result will be a  $p \times r$  matrix C. There are  $(p \cdot r)$  total entries in C and each takes \_\_\_\_\_ to compute.

**A.  $O(q)$**

B.  $O(1)$

C.  $O(n^2)$

D.  $O(n^3)$

**Question No:84**

**(Marks:01)**

**Vu-Topper RM**

One of the clever aspects of heaps is that they can be stored in arrays without using any -----.

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**A. Pointers**

- B. constants
- C. variables
- D.functions

**Question No:85 (Marks:01)**

**Vu-Topper RM**

Merge sort requires extra array storage,

**A. True**

B.False

**Question No:86 (Marks:01)**

**Vu-Topper RM**

The Huffman codes provide a method of encoding data **inefficiently** when coded using ASCII standard.

A. True

**B. False**

**Question No:87 (Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with \_\_\_ bits.

**A. 80**

B. 160

C. 320

D. 100

**Question No:88 (Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 160 bits.

A. True

**B. False**

**Question No:89 (Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 10 bytes.

A. True

**B. False**

**Question No:90 (Marks:01)**

**Vu-Topper RM**

The greedy part of the Huffman encoding algorithm is to first find two nodes with **character** frequency

A. True

**B. False**

**Question No:91 (Marks:01)**

**Vu-Topper RM**

Huffman algorithm uses a greedy approach to generate an prefix code T that minimizes the expected length  $B(T)$  of the encoded string.

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**A. True**

B. False

**Question No:92**

**(Marks:01)**

**Vu-Topper RM**

An optimization problem is one in which you want to find,

A. Not a solution

B. An algorithm

C. Good solution

**D. The best solution**

**Question No:93**

**(Marks:01)**

**Vu-Topper RM**

Although it requires more complicated data structures, Prim's algorithm for a minimum spanning tree is better than Kruskal's when the graph has a large number of vert

**A. True**

B.False

**Question No:94**

**(Marks:01)**

**Vu-Topper RM**

If a problem is in NP, it must also be in P.

A. True

B. False

**C.unknown**

**Question No:95**

**(Marks:01)**

**Vu-Topper RM**

What is generally true of Adjacency List and Adjacency Matrix representations of graphs?

**A. Lists require less space than matrices but take longer to find the weight of an edge (v1,v2)**

B. Lists require less space than matrices and they are faster to find the weight of an edge (v1,v2)

C. Lists require more space than matrices and they take longer to find the weight of an edge (v1,v2)

D. Lists require more space than matrices but are faster to find the weight of an edge (v1,v2)

**Question No:96**

**(Marks:01)**

**Vu-Topper RM**

If a graph has v vertices and e edges then to obtain a spanning tree we have to delete

A. v edges.

B.  $v - e + 5$  edges

C.  $v + e$  edges.

**D. None of these**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:97**

**(Marks:01)**

**Vu-Topper RM**

Maximum number of vertices in a Directed Graph may be  $|V^2|$

- A. True
- B. False**

**Question No:98**

**(Marks:01)**

**Vu-Topper RM**

The Huffman algorithm finds a (n) \_\_solution.

- A. Optimal**
- B. Non-optimal
- C. Exponential
- D. Polynomial

**Question No:99**

**(Marks:01)**

**Vu-Topper RM**

Edge (u, v) is a forward edge if

- A. u is a proper descendant of v in the tree
- B. v is a proper descendant of u in the tree pg#129**
- C. None of these

**Question No:100**

**(Marks:01)**

**Vu-Topper RM**

In counting sort, once we know the ranks, we simply \_\_\_ numbers to their final positions in an output array.

- A. Delete
- B. copy**
- C. Mark
- D. arrange

**Question No:101**

**(Marks:01)**

**Vu-Topper RM**

Dynamic programming algorithms need to store the results of intermediate sub-problems.

- A. True**
- B. False

**Question No:102**

**(Marks:01)**

**Vu-Topper RM**

\_\_\_\_\_ is a graphical representation of an algorithm

- A. notation
- B. notation
- C. Flowchart**
- D. Asymptotic notation

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:103**

**(Marks:01)**

**Vu-Topper RM**

Which of the following is calculated with **big o notation**?

- A. Lower bounds
- B. Upper bounds**
- C. Both upper and lower bound
- D. Medium bounds

**Question No:104**

**(Marks:01)**

**Vu-Topper RM**

Merge sort makes two recursive calls. Which statement is true after these recursive calls finish, but before the merge step?

- A. The array elements form a heap
- B. Elements in each half of the array are sorted amongst themselves**
- C. Elements in the first half of the array are less than or equal to elements in the second half of the array
- D. None of the above

**Question No:105**

**(Marks:01)**

**Vu-Topper RM**

What is the solution to the recurrence  $T(n) = T(n/2) + n$ ,  $T(1) = 1$

- A.  $O(\log n)$
- B.  $O(n)$**
- C.  $O(n \log n)$
- D.  $O(2n)$

**Question No:106**

**(Marks:01)**

**Vu-Topper RM**

Consider the following Huffman Tree The binary code for the string TEA is

- A. 00 010**
- B. 011 00 010
- C. 10 00 110
- D. 11 10 110

**Question No:107**

**(Marks:01)**

**Vu-Topper RM**

A greedy algorithm does not work in phases.

- A. True
- B. False**

**Question No:108**

**(Marks:01)**

**Vu-Topper RM**

Can an adjacency matrix for a directed graph ever not be square in shape?

- A. Yes
- B. No**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:109**

**(Marks:01)**

**Vu-Topper RM**

One of the clever aspects of heaps is that they can be stored in arrays without using any\_\_\_\_\_.

**A. Pointers**

B. constants

C. variables

D.functions

**Question No:110**

**(Marks:01)**

**Vu-Topper RM**

Non-optimal or greedy algorithm for money change takes\_\_\_\_\_

**A. O(k)**

B. O(kN)

C. O(2k)

D. O(N)

**Question No:111**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 320 bits.

A. True

**B. False**

**Question No:112**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 100 bits.

A. True

**B. False**

**Question No:113**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 32 bytes

A. True

**B. False**

**Question No:114**

**(Marks:01)**

**Vu-Topper RM**

Huffman algorithm uses a greedy approach to generate an **antefix** code T that minimizes the expected length B (T) of the encoded string.

A. True

**B. False**

**Question No:115**

**(Marks:01)**

**Vu-Topper RM**

Depth first search is shortest path algorithm that works on un-weighted graphs.

A. True

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**B.False**

**Question No:116** (Marks:01) **Vu-Topper RM**  
Floyd-Warshall algorithm is a dynamic programming algorithm; the genius of the algorithm is in the clever recursive formulation of the shortest path problem.

**A. True**

B. Flase

**Question No:117** (Marks:01) **Vu-Topper RM**  
Floyd-Warshall algorithm, as in the case with DP algorithms, we avoid recursive evaluation by generating a table for

A. k

**B.  $d^k$**

C. True

D. Flase

**Question No:118** (Marks:01) **Vu-Topper RM**  
The term coloring came from the original application which was in map drawing.

**A. True**

B. False

**Question No:119** (Marks:01) **Vu-Topper RM**  
In the clique cover problem, for two vertices to be in the same group, they must be \_\_\_\_\_ each other.

A. Apart from

B. Far from

C. Near to

**D. Adjacent to**

**Question No:120** (Marks:01) **Vu-Topper RM**  
Fixed-length codes may not be efficient from the perspective of \_\_\_\_\_ the total quantity of data.

**A. Minimizing**

B. Averaging

C. Maximizing

D. Summing

**Question No:121** (Marks:01) **Vu-Topper RM**  
In greedy algorithm, at each phase, you take the \_\_\_\_\_ you can get right now, without regard for future consequences.

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- A. Worst
- B. Minimum
- C. Good
- D. Best**

**Question No:122** (Marks:01) **Vu-Topper RM**

The difference between Prim's algorithm and Dijkstra's algorithm is that Dijkstra's algorithm uses a same key.

- A. True
- B. False**

**Question No:123** (Marks:01) **Vu-Topper RM**

If there are n items, there are \_\_\_ possible combinations of the items.

- A. 2
- B. N
- C.  $2^n$**
- D.  $3^n$

**Question No:124** (Marks:01) **Vu-Topper RM**

In Knapsack Problem, the thief's goal is to put items in the bag such that the \_\_\_\_\_ of the items does not exceed the limit of the bag.

- A. Value**
- B. Weight
- C. Length
- D. Balance

**Question No:125** (Marks:01) **Vu-Topper RM**

The knapsack problem does not belong to the domain of optimization problems.

- A. True
- B. False**

**Question No:126** (Marks:01) **Vu-Topper RM**

In Huffman encoding, for a given message string, the frequency of occurrence (relative probability) of each character in the message is determined last.

- A. True
- B. False**

**Question No:127** (Marks:01) **Vu-Topper RM**

Fixed-length codes are known for easy break up of a string into its individual characters.

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**A. True**

B. False

**Question No:128**

**(Marks:01)**

**Vu-Topper RM**

In Knapsack Problem, value and weight both are to be under consideration.

**A. True**

B. False

**Question No:129**

**(Marks:01)**

**Vu-Topper RM**

Multiplication is\_\_.

A. log n

B. n

C. n<sup>2</sup>

**D. n<sup>3</sup>**

**Question No:130**

**(Marks:01)**

**Vu-Topper RM**

In DP based solution of knapsack problem, to compute entries of V we will imply a/an\_\_approach.

A. Subjective

**B. Inductive**

C.Brute force

D.Combination

**Question No:131**

**(Marks:01)**

**Vu-Topper RM**

A greedy algorithm sometimes works well for optimization problems.

**A. True**

B. False

**Question No:132**

**(Marks:01)**

**Vu-Topper RM**

In Huffman encoding, frequency of each character can be determined by parsing the message and \_how many times each character (or symbol) appears.

A. Printing

B. Incrementing

**C. Counting**

D. Deleting

**Question No:133**

**(Marks:01)**

**Vu-Topper RM**

Greedy algorithm can do very poorly for some problems.

**A. True**

B. False

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:134**

**(Marks:01)**

**Vu-Topper RM**

The Huffman codes provide a method of \_\_\_ data efficiently.

- A. Reading
- B. Encoding**
- C. Decoding
- D. Printing

**Question No:135**

**(Marks:01)**

**Vu-Topper RM**

In \_\_\_ based solution of knapsack problem, we consider 2 cases, Leave object Or Take object.

- A. Brute force
- B. Dynamic programming**

**Question No:136**

**(Marks:01)**

**Vu-Topper RM**

In brute force based solution of knapsack problem, we consider 2 cases, Leave object Or Take object.

- A. TRUE
- B. False**

**Question No:137**

**(Marks:01)**

**Vu-Topper RM**

\_\_\_ problem, we want to find the best solution.

- A. Minimization
- B. Averaging
- C. Optimization**
- D. Maximization

**Question No:138**

**(Marks:01)**

**Vu-Topper RM**

Counting Money problem is an example which cannot be optimally solved by greedy algorithm.

- A. True**
- B. False

**Question No:139**

**(Marks:01)**

**Vu-Topper RM**

Huffman algorithm generates an optimum prefix code.

- A. True**
- B. False

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:140**

**(Marks:01)**

**Vu-Topper RM**

If the string "lmncde" is coded with ASCII code, the message length would be \_\_\_bits.

A. 24

B. B36

**C. 48**     $6*8=48$

D. 60

**Question No:141**

**(Marks:01)**

**Vu-Topper RM**

There are \_\_\_nested loops in DP based algorithm for computing the minimum cost of chain matrix multiplication.

A. 2

**B. 3**

C. 4

**Question No:142**

**(Marks:01)**

**Vu-Topper RM**

A number of lectures are to be given in a single lecture hall. Optimum scheduling for this is an example of Activity selection.

**A. True**

B. False

**Question No:143**

**(Marks:01)**

**Vu-Topper RM**

The activity scheduling is a simple scheduling problem for which the greedy algorithm approach provides a/an \_\_\_solution.

A. Simple

B. Sub optimal

**C. Optimal**

D. Non optimal

**Question No:144**

**(Marks:01)**

**Vu-Topper RM**

The string |xyz|, if coded with ASCII code, the message length would be 24 bits.

**A. True** ( $3*8=24$ )

B. False

**Question No:145**

**(Marks:01)**

**Vu-Topper RM**

An application problem is one in which you want to find, not just a solution, but the \_\_\_solution.

A. Simple

**B. Good**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

C. Best

**Question No:146**

**(Marks:01)**

**Vu-Topper RM**

Suppose that a graph  $G = (V, E)$  is implemented using adjacency lists. What is the complexity of a breadth-first traversal of  $G$ ?

- A.  $O(|V|^2)$
- B.  $O(|V| + |E|)$
- C.  $O(|V|^2|E|)$
- D.  $O(|V| + |E|)$**

**Question No:147**

**(Marks:01)**

**Vu-Topper RM**

Which is true statement?

- A. Breadth first search is shortest path algorithm that works on un-weighted graphs**
- B. Depth first search is shortest path algorithm that works on un-weighted graphs.
- C. Both of above are true.
- D. None of above are true.

**Question No:148**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string "abacdaacacwe" will be encoded with \_\_\_\_\_ bits

- A. 64
- B. 128**
- C. 96
- D. 120

**Question No:149**

**(Marks:01)**

**Vu-Topper RM**

**Question No:150**

**(Marks:01)**

**Vu-Topper RM**

**Question No:151**

**(Marks:01)**

**Vu-Topper RM**

**Question No:152**

**(Marks:01)**

**Vu-Topper RM**

**Question No:153**

**(Marks:01)**

**Vu-Topper RM**

Kruskal's algorithm (choose best non-cycle edge) is better than Prim's (choose best tree edge) when the graph has relatively few edges.

- A. True**
- B. False

**Question No:154**

**(Marks:01)**

**Vu-Topper RM**

What algorithm technique is used in the implementation of Kruskal solution for the MST?

- A. Greedy Technique**
- B. Divide-and-Conquer Technique

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

C. Dynamic Programming Technique

D. The algorithm combines more than one of the above techniques

**Question No:155** (Marks:01)

**Vu-Topper RM**

**Question No:156** (Marks:01)

**Vu-Topper RM**

In Prim's algorithm, we will make use of\_\_\_\_\_.

**A. Priority Queue**

**Question No:157** (Marks:01)

**Vu-Topper RM**

**Question No:158** (Marks:01)

**Vu-Topper RM**

What is the time complexity to extract a vertex from the priority queue in Prim's algorithm?

A.  $O(\log E)$

B.  $(V)$

C.  $(V+E)$

**D.  $O(\log V)$**

**Question No:159** (Marks:01)

**Vu-Topper RM**

**Question No:160** (Marks:01)

**Vu-Topper RM**

**Question No:161** (Marks:01)

**Vu-Topper RM**

There is relationship between number of back edges and number of cycles in DFS

A. Both are equal.

B. Cycles are half of back edges.

C. Cycles are one fourth of back edges.

**D. There is no relationship between back edges and number of cycles.**

**Question No:162** (Marks:01)

**Vu-Topper RM**

**Question No:163** (Marks:01)

**Vu-Topper RM**

In in-place sorting algorithm is one that uses arrays for storage :

A. An additional array

**B. No additional array**

C. Both of above may be true according to algorithm

D. More than 3 arrays of one dimension.

**Question No:164** (Marks:01)

**Vu-Topper RM**

In stable sorting algorithm

A. One array is used

B. In which duplicating elements are not handled.

C. More than one arrays are required.

**D. Duplicating elements remain in same relative position after sorting.**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:165 (Marks:01)**

**Vu-Topper RM**

Which sorting algorithm is faster :

- A.  $O(n^2)$
- B.  $O(n \log n)$**
- C.  $O(n+k)$
- D.  $O(n^3)$

**Question No:166 (Marks:01)**

**Vu-Topper RM**

In Quick sort algorithm, constants hidden in  $T(n \lg n)$  are

- A. Large
- B. Medium
- C. Not known
- D. Small**

**Question No:167 (Marks:01)**

**Vu-Topper RM**

Quick sort is based on divide and conquer paradigm; we divide the problem on base of pivot element and:

- A. There is explicit combine process as well to conquer the solution.
- B. No work is needed to combine the sub-arrays, the array is already sorted
- C. Merging the sub arrays
- D. None of the above**

**Question No:168 (Marks:01)**

**Vu-Topper RM**

Dijkstra's algorithm :

- A. Has greedy approach to find all shortest paths
- B. Has both greedy and Dynamic approach to find all shortest paths
- C. Has greedy approach to compute single source shortest paths to all other vertices**
- D. Has both greedy and dynamic approach to compute single source shortest paths to all other vertices.

**Question No:169 (Marks:01)**

**Vu-Topper RM**

Which may be stable sort:

- A. Bubble sort
- B. Insertion sort
- C. Selection sort
- D. Both of above**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:170**

**(Marks:01)**

**Vu-Topper RM**

In the analysis of Selection algorithm, we eliminate a constant fraction of the array with each phase; we get the convergent \_\_ series in the analysis,

- A. linear
- B. arithmetic
- C. exponent
- D. Geometric**

**Question No:171**

**(Marks:01)**

**Vu-Topper RM**

Which may be stable sort:

- A. Bubble sort
- B. Insertion sort
- C. Selection sort
- D. Both of above**

**Question No:172**

**(Marks:01)**

**Vu-Topper RM**

In the analysis of Selection algorithm, we eliminate a constant fraction of the array with each phase; we get the convergent \_\_ series in the analysis,

- A. linear
- B. arithmetic
- C. exponent
- D. Geometric**

**Question No:173**

**(Marks:01)**

**Vu-Topper RM**

How much time merge sort takes for an array of numbers?

- A.  $T(n^2)$
- B.  $T(n)$**
- C.  $T(\log n)$
- D.  $T(n \log n)$

**Question No:174**

**(Marks:01)**

**Vu-Topper RM**

We do sorting to,

- A. keep elements in random positions
- B. keep the algorithm run in linear order
- C. keep the algorithm run in  $(\log n)$  order
- D. keep elements in increasing or decreasing order**

**Question No:175**

**(Marks:01)**

**Vu-Topper RM**

Dynamic programming algorithms need to store the results of intermediate sub-problems.

- A. True**
- B. False

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:176** (Marks:01) **Vu-Topper RM**

Dijkstras single source shortest path algorithm works if all edges weights are non negative and there are no negative cost cycles.

A. False

**B. True**

**Question No:177** (Marks:01) **Vu-Topper RM**

Dijkstra s single source shortest path algorithm works if all edges weights are negative and there are no negative cost cycles.

A. True

**B. False**

**Question No:178** (Marks:01) **Vu-Topper RM**

In the clique cover problem, for two vertices to be in the same group, they must be \_\_\_ each other.

A. Apart from

B. Far from

C. Near to

**D. Adjacent to**

**Question No:179** (Marks:01) **Vu-Topper RM**

Fixed-length codes may not be efficient from the perspective of \_\_\_ the total quantity of data.

**A. Minimizing**

B. Averaging

C. Maximizing

D. Summing

**Question No:180** (Marks:01) **Vu-Topper RM**

In \_\_\_ based solution of knapsack problem, we consider 2 cases, Leave object Or Take object.

A. Brute force

**B. Dynamic programming**

**Question No:181** (Marks:01) **Vu-Topper RM**

In brute force based solution of knapsack problem, we consider 2 cases, Leave object Or Take object.

A. TRUE

**B. False**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:182**

**(Marks:01)**

**Vu-Topper RM**

\_\_\_\_\_problem, we want to find the best solution.

- A. Minimization
- B. Averaging
- C. Optimization**
- D. Maximization

**Question No:183**

**(Marks:01)**

**Vu-Topper RM**

Using ASCII standard the string abacdaacac will be encoded with 10 bytes.

- A. True**
- B. False

**Question No:184**

**(Marks:01)**

**Vu-Topper RM**

How many elements do we eliminate in each time for the Analysis of Selection algorithm?

- A.  $(n / 2) + n$  elements
- B.  $n / 4$  elements
- C.  $2n$  elements
- D.  $n / 2$  elements**

**Question No:185**

**(Marks:01)**

**Vu-Topper RM**

Slow sorting algorithms run in,

- A.  $T(n^2)$**
- B.  $T(n)$
- C.  $T(\log n)$
- D.  $T(n \log n)$

**Question No:186**

**(Marks:01)**

**Vu-Topper RM**

Counting sort is suitable to sort the elements in range 1 to k:

- A. K is large
- B. K is small**
- C. K may be large or small
- D. None

**Question No:187**

**(Marks:01)**

**Vu-Topper RM**

Heaps can be stored in arrays without using any pointers; this is due to the \_\_\_\_\_ nature of the binary tree,

- A. left-complete**
- B. right-complete
- C. tree nodes
- D. tree leaves

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:188**

**(Marks:01)**

**Vu-Topper RM**

A heap is a left-complete binary tree that conforms to the \_\_\_\_\_

- A. increasing order only
- B. heap order**
- C. decreasing order only
- D.  $(\log n)$  order

**Question No:189**

**(Marks:01)**

**Vu-Topper RM**

Divide-and-conquer as breaking the problem into a small number of

- A. pivot
- B. Sieve
- C. smaller sub problems**
- D. Selection

**Question No:190**

**(Marks:01)**

**Vu-Topper RM**

In Sieve Technique we do not know which item is of interest

- A. True**
- B. False

**Question No:191**

**(Marks:01)**

**Vu-Topper RM**

The recurrence relation of Tower of Hanoi is given below  $T(n) = \{ 1 \text{ if } n=1 \text{ and } 2T(n-1) \text{ if } n > 1$  In order to move a tower of 5 rings from one peg to another, how many ring moves are required?

- A. 16
- B. 10
- C. 32
- D. 31**

**Question No:192**

**(Marks:01)**

**Vu-Topper RM**

For the heap sort, access to nodes involves simple \_\_\_\_\_ operations.

- A. arithmetic**
- B. binary
- C. algebraic
- D. logarithmic

**Question No:193**

**(Marks:01)**

**Vu-Topper RM**

For the sieve technique we solve the problem,

- A. recursively**
- B. mathematically
- C. precisely
- D. accurately

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:194** (Marks:01)

**Vu-Topper RM**

The sieve technique works in\_ as follows

- A. phases**
- B. numbers
- C. integers
- D. routines

**Question No:195** (Marks:01)

**Vu-Topper RM**

A (an)\_\_\_\_\_ is a left-complete binary tree that conforms to the heap order

- A. heap**
- B. binary tree
- C. binary search tree
- D. array

**Question No:196** (Marks:01)

**Vu-Topper RM**

The sieve technique is a special case, where the number of sub problems is just

- A. 5
- B. Many
- C. 1**
- D. Few

**Question No:197** (Marks:01)

**Vu-Topper RM**

Analysis of Selection algorithm ends up with,

- A.  $T(n)$
- B.  $T(1 / 1 + n)$
- C.  $T(n / 2)$
- D.  $T((n / 2) + n)$**

**Question No:198** (Marks:01)

**Vu-Topper RM**

For the heap sort we store the tree nodes in

- A. level-order traversal**
- B. in-order traversal
- C. pre-order traversal
- D. post-order traversal

**Question No:199** (Marks:01)

**Vu-Topper RM**

The reason for introducing Sieve Technique algorithm is that it illustrates a very important special case of,

- A. divide-and-conquer**
- B. decrease and conquer

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- C. greedy nature
- D. 2-dimension Maxima

**Question No:200** (Marks:01) **Vu-Topper RM**

Sieve Technique applies to problems where we are interested in finding a single item from a larger set of \_\_\_\_\_

- A. n items**
- B. phases
- C. pointers
- D. constant

**Question No:201** (Marks:01) **Vu-Topper RM**

Memorization is?

- A. To store previous results for future use
- B. To avoid this unnecessary repetitions by writing down the results of recursive calls and looking them up again if we need them later**
- C. To make the process accurate
- D. None of the above

**Question No:202** (Marks:01) **Vu-Topper RM**

Quick sort is

- A. Not stable but in place**
- B. Stable & in place
- C. Stable but not in place
- D. Some time stable & some times in place

**Question No:203** (Marks:01) **Vu-Topper RM**

One example of in place but not stable algorithm is

- A. Merger Sort
- B. Quick Sort**
- B. Continuation Sort
- C. Bubble Sort

**Question No:204** (Marks:01) **Vu-Topper RM**

Continuation sort is suitable to sort the elements in range 1 to k

- A. K is Large
- B. K is not known
- C. K may be small or large
- D. K is small**

**Question No:205** (Marks:01) **Vu-Topper RM**

Which may be a stable sort?

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- A. Merger
- B. Insertion
- C. None of the above
- D. Both above**

**Question No:206** (Marks:01) **Vu-Topper RM**

An in place sorting algorithm is one that uses \_\_\_\_\_ arrays for storage

- A. Two dimensional arrays
- B. More than one array
- C. No Additional Array**
- D. None of the above

**Question No:207** (Marks:01) **Vu-Topper RM**

Continuing sort has time complexity of ?

- A.  $O(n)$**
- B.  $O(n+k)$
- C.  $O(n \log n)$
- D.  $O(k)$

**Question No:208** (Marks:01) **Vu-Topper RM**

single item from a larger set of \_\_\_\_\_

- A. phases
- B. n items**
- C. pointers
- D. constant

**Question No:209** (Marks:01) **Vu-Topper RM**

For the Sieve Technique we take time

- A.  $T(nk)$**
- B.  $T(n/3)$
- C.  $n^2$
- D.  $n/3$

**Question No:210** (Marks:01) **Vu-Topper RM**

Due to left complete nature of binary tree, the heap can be stored in

- A. Arrays**
- B. Structures
- C. Link List
- D. Stack

**Question No:211** (Marks:01) **Vu-Topper RM**

What type of instructions Random Access Machine (RAM) can execute?

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- A. Algebraic and logic
- B. Geometric and arithmetic
- C. Arithmetic and logic**
- D. Parallel and recursive

**Question No:212 (Marks:01)**

**Vu-Topper RM**

What is the total time to heapify?

- A.  $O(\log n)$**
- B.  $O(n \log n)$
- C.  $O(n^2 \log n)$
- D.  $O(\log^2 n)$

**Question No:213 (Marks:01)**

**Vu-Topper RM**

word Algorithm comes from the name of the muslim author \_\_\_\_

- A. Abu Ja'far Mohammad ibn Musa al-Khowarizmi.**

**Question No:214 (Marks:01)**

**Vu-Topper RM**

Al-Khwarizmi's work was written in a book titled \_\_\_\_\_

- A. al Kitab al-mukhtasar fi hisab al-jabr wa'l-muqabalah**

**Question No:215 (Marks:01)**

**Vu-Topper RM**

Random access machine or RAM is a/an

- A. Machine build by Al-Khwarizmi
- B. Mechanical machine
- C. Electronics machine
- D. Mathematical model**

**Question No:216 (Marks:01)**

**Vu-Topper RM**

A RAM is an idealized machine with\_ random-access memory.

- A. 256MB
- B. 512MB
- C. 100GB
- D. an infinitely large**

**Question No:217 (Marks:01)**

**Vu-Topper RM**

What will be the total number of max comparisons if we run brute-force maxima algorithm with n elements?

- A.  $2n$
- B.  $2nn$
- C.  $n$**
- D.  $8n$

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:218**

**(Marks:01)**

**Vu-Topper RM**

```
Consider the following code:For(j=1; j<n;j++)
For(k=1; k<15;k++)
For(l=5; l<n; l++)
{
Do_something_constant();
}
```

What is the order of execution for this code.

- A.  $O(n)$**
- B.  $O(n^3)$
- C.  $O(n^2 \log n)$
- D.  $O(n^2)$

**Question No:219**

**(Marks:01)**

**Vu-Topper RM**

Is it possible to sort without making comparisons?

- A. Yes**
- B. No

**Question No:220**

**(Marks:01)**

**Vu-Topper RM**

When we call heapify then at each level the comparison performed takes time

- A. It will take  $\Theta(1)$**
- B. Time will vary according to the nature of input data
- C. It can not be predicted
- D. It will take  $\Theta(\log n)$

**Question No:221**

**(Marks:01)**

**Vu-Topper RM**

In Quick sort, we don't have the control over the sizes of recursive calls

- A. True**
- B. False
- C. Less information to decide
- D. Either true or false

**Question No:222**

**(Marks:01)**

**Vu-Topper RM**

For Chain Matrix Multiplication we cannot use divide and conquer approach because,

- A. We do not know the optimum k**
- B. We use divide and conquer for sorting only
- C. We can easily perform it in linear time
- D. Size of data is not given

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:223**

**(Marks:01)**

**Vu-Topper RM**

Suppose we have three items as shown in the following table, and suppose the capacity of the knapsack is 50 i.e.  $W = 50$ .

Item	Value	Weight
1	60	10
2	100	20
3	120	30

The optimal solution is to pick

- A. Items 1 and 2
- B. Items 1 and 3
- C. Items 2 and 3**
- D. None of these

**Question No:224**

**(Marks:01)**

**Vu-Topper RM**

who invented the quick sort

- A. C.A.R. Hoare**

**Question No:225**

**(Marks:01)**

**Vu-Topper RM**

main elements to a divide-and-conquer

- A. Divide, conquer, combine**

**Question No:226**

**(Marks:01)**

**Vu-Topper RM**

Mergesort is a stable algorithm but not an in-place algorithm.

- A. True**
- B. False

**Question No:227**

**(Marks:01)**

**Vu-Topper RM**

Counting sort the numbers to be sorted are in the range 1 to k where k is small.

- A. True**
- B. False

**Question No:228**

**(Marks:01)**

**Vu-Topper RM**

In selection algorithm, because we eliminate a constant fraction of the array with each phase, we get the

- A. Convergent geometric series**
- B. Divergent geometric series
- C. None of these

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:229** (Marks:01)

**Vu-Topper RM**

In RAM model instructions are executed

- A. One after another**
- B. Parallel

**Question No:230** (Marks:01)

**Vu-Topper RM**

Due to left-complete nature of binary tree, heaps can be stored in

- A. Link list
- B. Structure
- C. Array**
- D. None of above

**Question No:231** (Marks:01)

**Vu-Topper RM**

The time assumed for each basic operation to execute on RAM model of computation is-----

- A. Constant**
- B. Infinite
- C. Continuous
- D. Variable

**Question No:232** (Marks:01)

**Vu-Topper RM**

If the indices passed to merge sort algorithm are not equal, the algorithm may return immediately.

- A. True
- B.False**

**Question No:233** (Marks:01)

**Vu-Topper RM**

Brute-force algorithm uses no intelligence in pruning out decisions.

- A. True**
- B. False

**Question No:234** (Marks:01)

**Vu-Topper RM**

In analysis, the Upper Bound means the function grows asymptotically no faster than its largest term.

- A. True**
- B. False

**Question No:235** (Marks:01)

**Vu-Topper RM**

For small values of n, any algorithm is fast enough. Running time does become an issue when n gets large

- A. True .**
- B.False

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:236** (Marks:01) **Vu-Topper RM**

In simple brute-force algorithm, we give no thought to efficiency.

- A. True**
- B. False

**Question No:237** (Marks:01) **Vu-Topper RM**

The ancient Roman politicians understood an important principle of good algorithm design that is plan-sweep algorithm.

- A. True
- B. False**

**Question No:238** (Marks:01) **Vu-Topper RM**

If the indices passed to merge sort algorithm are \_\_\_\_, then this means that there is only one element to sort.

- A. Small
- B. Large
- C. Equal**
- D. Not Equal

**Question No:239** (Marks:01) **Vu-Topper RM**

In pseudo code, the level of details depends on intended audience of the algorithm.

- A. True**
- B. False

**Question No:240** (Marks:01) **Vu-Topper RM**

In 2d-space a point is said to be \_\_\_\_\_ if it is not dominated by any other point in that space.

- A. Member
- B. Minimal
- C. Maximal**
- D. Joint

**Question No:241** (Marks:01) **Vu-Topper RM**

An algorithm is a mathematical entity that is dependent on a specific programming language.

- A. True
- B. False**

**Question No:242** (Marks:01) **Vu-Topper RM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

The running time of an algorithm would not depend upon the optimization by the compiler but that of an implementation of the algorithm would depend on it.

**A. TRUE**

B. False

**Question No:243**

**(Marks:01)**

**Vu-Topper RM**

$F(n)$  and  $g(n)$  are asymptotically equivalent. This means that they have essentially the same \_\_\_\_\_ for large  $n$ .

A. Results

B. Variables

**C. Growth rates**

D. Size

**Question No:244**

**(Marks:01)**

**Vu-Topper RM**

$8n^2 + 2n - 3$  will eventually exceed  $c^2 \cdot (n)$  no matter how large we make  $c^2$ .

**A. TRUE**

B. False

**Question No:245**

**(Marks:01)**

**Vu-Topper RM**

If we associate  $(x, y)$  integers pair to cars where  $x$  is the speed of the car and  $y$  is the negation of the price. High  $y$  value for a car means a \_\_ car.

A. Fast

B. Slow

**C. Cheap**

D. Expensive

**Question No:246**

**(Marks:01)**

**Vu-Topper RM**

While solving Selection problem, in Sieve technique we partition input data

A. In increasing order

B. In decreasing order

**C. According to Pivot**

D. Randomly

**Question No:247**

**(Marks:01)**

**Vu-Topper RM**

In Sieve Technique, we know the item of interest.

A. True

**B. False**

**Question No:248**

**(Marks:01)**

**Vu-Topper RM**

In Heap Sort algorithm, we build \_\_\_\_\_ for ascending sort.

**A. Max heap**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

B. Min heap

**Question No:249**

**(Marks:01)**

**Vu-Topper RM**

Quick sort is best from the perspective of Locality of reference.

**A. True**

B. False

**Question No:250**

**(Marks:01)**

**Vu-Topper RM**

While Sorting, the ordered domain means for any two input elements x and y \_\_\_\_\_ satisfies only.

A.  $x < y$

B.  $x > y$

C.  $x = y$

**D. All of the above**

**Question No:251**

**(Marks:01)**

**Vu-Topper RM**

Algorithm is a mathematical entity, which is independent of a specific machine and operating system.

A. True

**B. False**

**Question No:252**

**(Marks:01)**

**Vu-Topper RM**

In Heap Sort algorithm, the total running time for Heapify procedure is

**A. Theta ( $\log n$ )**

B. Order ( $\log n$ )

C. Omega ( $\log n$ )

D. O (1)

**Question No:253**

**(Marks:01)**

**Vu-Topper RM**

A point p in 2-dimensional space is usually given by its integer coordinate(s)\_

A. p.x only p.y

B. only p.x & p.z

**C. p.x & p.y**

**Question No:254**

**(Marks:01)**

**Vu-Topper RM**

In Heap Sort algorithm, the maximum levels an element can move upward is

**A. Theta ( $\log n$ )**

B. Order ( $\log n$ )

C. Omega ( $\log n$ )

D. O

**Question No:255**

**(Marks:01)**

**Vu-Topper RM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

Floor and ceiling are \_\_\_\_\_ to calculate while analyzing algorithms.

A. Very easy

**B. Usually considered difficult**

**Question No:256 (Marks:01)**

**Vu-Topper RM**

\_\_\_\_\_ is one of the few problems, where provable lower bounds exist on how fast we can sort.

A. Searching

**B. Sorting**

C. Both Searching & Sorting

D. Graphing

**Question No:257 (Marks:01)**

**Vu-Topper RM**

A RAM is an idealized algorithm with takes an infinitely large random-access memory.

A. True

**B. False**

**Question No:258 (Marks:01)**

**Vu-Topper RM**

Upper bound requires that there exist positive constants  $c_2$  and  $n_0$  such that  $f(n) \leq c_2 n$  for all  $n \leq n_0$  (ye question ghalat lag raha hai mujhae)

A. Less than

**B. Equal to or Less than**

C. Equal or Greater than

D. Greater than

**Question No:259 (Marks:01)**

**Vu-Topper RM**

In Heap Sort algorithm, if heap property is violated \_\_\_\_\_

A. We call Build heap procedure

**B. We call Heapify procedure**

C. We ignore

D. Heap property can never be violated

**Question No:260 (Marks:01)**

**Vu-Topper RM**

In \_\_\_\_\_ we have to find rank of an element from given input.

A. Merge sort algorithm

**B. Selection problem**

C. Brute force technique

D. Plane Sweep algorithm

**Question No:261 (Marks:01)**

**Vu-Topper RM**

A point  $p$  in 2-dimensional space is usually given by its integer coordinate(s) \_\_\_\_\_

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- A. p.x only
- B. p.y only
- C. p.x & p.z
- D. p.x & p.y**

**Question No:262** (Marks:01) **Vu-Topper RM**

In Quick Sort Constants hidden in  $T(n \log n)$  are

- A. Large
- B. Medium
- C. Small**
- D. Not Known

**Question No:263** (Marks:01) **Vu-Topper RM**

The running time of quick sort depends heavily on the selection of

- A. No of inputs
- B. Arrangement of elements in array
- C. Size o elements
- D. Pivot elements**

**Question No:264** (Marks:01) **Vu-Topper RM**

Choose one Counting sort has timecomplexity:

- A.  $O(n)$  (Page 58)**
- B.  $O(n+k)$
- C.  $O(k)$
- D.  $O(n \log n)$

**Question No:265** (Marks:01) **Vu-Topper RM**

In place stable sorting algorithm.

- A. If duplicate elements remain in the same relative position after sorting**
- B. One array is used
- C. More than one arrays are required
- D. Duplicating elements not handled

**Question No:266** (Marks:01) **Vu-Topper RM**

Cont sort is suitable to sort the elements in range 1 to k

- A. K is Large
- B. K is not known
- C. K may be small or large
- D. K is small**

**Question No:267** (Marks:01) **Vu-Topper RM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

The sieve technique works where we have to find \_\_item(s) from a large input.

**A. Single**

B. Two

C. Three

D. Similar

**Question No:268**

**(Marks:01)**

**Vu-Topper RM**

A RAM is an idealized machine with \_\_random-access memory.

A. 256MB

B. 512MB

**C. an infinitely large**

D. 100GB

**Question No:269**

**(Marks:01)**

**Vu-Topper RM**

A RAM is an idealized machine with \_\_random-access memory.

A. 256MB

B. 512MB

**C. an infinitely large**

D. 100GB

**Question No:270**

**(Marks:01)**

**Vu-Topper RM**

For the sieve technique we solve the problem,

**A. recursively**

B. mathematically

C. precisely

D. accurately

**Question No:271**

**(Marks:01)**

**Vu-Topper RM**

number of nodes in a complete binary tree of height h is

**A.  $2^{(h+1)} - 1$**

B.  $2 * (h+1) - 1$

C.  $2 * (h+1)$

D.  $((h+1)^2) - 1$

**Question No:272**

**(Marks:01)**

**Vu-Topper RM**

Heaps can be stored in arrays without using any pointers; this is due to the \_\_\_\_\_ nature of the binary tree,

**A. left-complete**

B. right-complete

C. tree nodes

D. tree leaves

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:273**

**(Marks:01)**

**Vu-Topper RM**

Sorting is one of the few problems where provable\_bonds exists on how fast we can sort,

- A. upper
- B. lower**
- C. average
- D. log n

**Question No:274**

**(Marks:01)**

**Vu-Topper RM**

The analysis of Selection algorithm shows the total running time is indeed \_\_\_\_\_ in no

- A. arithmetic
- B. geometric
- C. linear**
- D. orthogonal

**Question No:275**

**(Marks:01)**

**Vu-Topper RM**

What algorithm technique is used in the implementation of Kruskal solution for the MST?

- A. Greedy Technique**
- B. Divide-and-Conquer Technique
- C. Dynamic Programming Technique
- D. The algorithm combines more than one of the above techniques

**Question No:276**

**(Marks:01)**

**Vu-Topper RM**

one If you find yourself in maze the better traversal approach will be :

- A. BFS**
- B. BFS and DFS both are valid
- C. Level order
- D. DFS

**Question No:277**

**(Marks:01)**

**Vu-Topper RM**

**Question No:278**

**(Marks:01)**

**Vu-Topper RM**

Suppose that a graph  $G = (V, E)$  is implemented using adjacency lists.

What is the complexity of a breadth-first traversal of  $G$ ?

- A.  $O(|V|^2)$
- B.  $O(|V| + |E|)$
- C.  $O(|V|^2|E|)$
- D.  $O(|V| + |E|)$**

**Question No:279**

**(Marks:01)**

**Vu-Topper RM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

There are\_\_nested loops in DP based algorithm for computing the minimum cost of chain matrix multiplication.

- A. 2
- B. 3**
- C. 4
- D. 5

**Question No:280** (Marks:01) **Vu-Topper RM**

Counting Money problem is an example which cannot be optimally solved by greedy algorithm.

- A. True**
- B. False

**Question No:281** (Marks:01) **Vu-Topper RM**

Who invented Quick sort procedure?

- A. Hoare**
- B. Sedgewick
- C. Mellroy
- D. Coreman

**Question No:282** (Marks:01) **Vu-Topper RM**

An activity scheduling algorithm, the width of a rectangle--.

- A. Is always ignored
- B. Direct toward recursion
- C. Should be maximized
- D. Indicates the duration of an activity**

**Question No:283** (Marks:01) **Vu-Topper RM**

In recursive formulation of Knapsack Problem:  $V[0, j] =$  for  $j \geq 0$

- A. -1
- B. 0**
- C. 1
- D. 2

**Question No:284** (Marks:01) **Vu-Topper RM**

The Probability in Huffman encoding is the number of occurrences of a character divided by the total-----in the message.

- A. Numbers
- B. Frequencies
- C. Strings
- D. Characters**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:285** (Marks:01)

**Vu-Topper RM**

Graphs can be represented by an----- and --.

- A. queue , stack
- B.adjacency list , adjacency matrix**
- C.adjacency right , adjacency left
- D.Binary , linear

**Question No:286** (Marks:01)

**Vu-Topper RM**

Edge weights can be interpreted as distance\_.

- A. in breadth-First Search
- B. in Queue's
- C. in the shortest-paths**
- D. in depth-First Search

**Question No:287** (Marks:01)

**Vu-Topper RM**

\_\_\_\_\_algorithm allows negative weights edges and no negative costcycles.

- A. Brute-force technique
- B. Bellman-Ford**
- C. Dijkstra's
- D. Print

**Question No:288** (Marks:01)

**Vu-Topper RM**

According to parenthesis lemma vertex u is a descendent of v vertex if and only if:

- A.  $f [d[u], f[u]] \subseteq [d[v], f[v]]$  (Page 129)**

**Question No:289** (Marks:01)

**Vu-Topper RM**

**Question No:290** (Marks:01)

**Vu-Topper RM**

Keeping in mind the shortest-path, if given scenarios occur in computer networks like the internet where data packets have to be routed. The vertices are\_\_\_\_\_and Edges are \_\_\_\_\_which may be wired or wireless.

- A. Routers, communication links**

**Question No:291** (Marks:01)

**Vu-Topper RM**

**Question No:292** (Marks:01)

**Vu-Topper RM**

Bellman-Ford algorithm is slower than\_\_\_\_\_.

- A. Dijkstra's**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:293**

**(Marks:01)**

**Vu-Topper RM**

From the given option's which one is correct regarding the time complexity of Dijkstra's algorithm:

**A.  $O(N^2)$**

**Question No:294**

**(Marks:01)**

**Vu-Topper RM**

In the shortest-paths problem, we are given a weighted of \_\_\_\_\_  $G = (V, E)$ .

**A. Un-directed graph**

**Question No:295**

**(Marks:01)**

**Vu-Topper RM**

**Question No:296**

**(Marks:01)**

**Vu-Topper RM**

**Question No:297**

**(Marks:01)**

**Vu-Topper RM**

In strong components algorithm, first of all DFS is run for computing finish times of vertices.

A.  $[d[u], f[u]] \subseteq [d[v], f[v]]$

B.  $[d[u], f[u]] \supseteq [d[v], f[v]]$

**C. unrelated**

D. Disjoint

**Question No:298**

**(Marks:01)**

**Vu-Topper RM**

**Question No:299**

**(Marks:01)**

**Vu-Topper RM**

What general property of the list indicates that the graph has an isolated vertex?

A. There is Null pointer at the end of list.

**B. The Isolated vertex is not handled in list.**

C. Only one value is entered in the list.

D. There is at least one null list.

**Question No:300**

**(Marks:01)**

**Vu-Topper RM**

As the Kruskal's runs, the edges in viable set A induce a \_\_\_\_\_ on the vertices.

A. Set

B. Graph

C. Tree

**D. Fore**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Visit My YouTube Channel**  
**For More Important Notes**  
**Channel Name = #VuTopperRM**

پری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365