Question # 1
Word Algorithm comes from the name of the Muslim author:
Abu Ja’far Mohammad ibn Musaal-Khowarizmi. (p7)

Question # 2
Al-Khwarizmi’s work was written in a book titled
al Kitab al-mukhatasar fi hisab al-jabr wa’l-muqabalah (p7)

Question # 3
Analysis of Selection algorithm ends up with,
Select correct option:
- T(n) (p37)
- T(1 / 1 + n)
- T(n / 2)
- T((n / 2) + n)

Question # 4
The function f(n)= n(logn+1)/2 is asymptotically equivalent to n log n. Here Upper Bound means the function f(n) grows asymptotically ________ faster than n log n.
Select correct option:
- More
- Quiet
- Not (p24)
- At least

Question # 5
The running time of an algorithm would not depend upon the optimization by the compiler but that of its implementation of the algorithm would depend on it.
Select correct option:
- True
- False (p7)

Question # 6
After sorting in merge sort algorithm, merging process is invoked.
Select correct option:
- True (p27)
- False

Question # 7
Asymptotic growth rate of the function is taken over_________ case running time.
Select correct option:
- Best
- Average
- Worst (p14 Note sure)
- Normal

Question # 8
In analysis of $f(n)=n(n/5)+n-10\log n$, $f(n)$ is asymptotically equivalent to ________.
Select correct option:
- $n$
- $2n$
- $n+1$
- $n^2$

**Question #9**

$f(n)$ and $g(n)$ are asymptotically equivalent. This means that they have essentially the same __________ for large $n$.
Select correct option:
- Results
- Variables
- Size
- Growth rates

**Question #10**

Algorithm is concerned with......issues.
Select correct option:
- Macro
- Micro
- Both Macro & Micro
- Normal

**Question #11**

We can not make any significant improvement in the running time which is better than that of brute-force algorithm.
Select correct option:
- True
- False

**Question #12**

In addition to passing in the array itself to Merge Sort algorithm, we will pass in ________other arguments which are indices.
Select correct option:
- Two
- Three
- Four
- Five

**Question #13**

For the worst-case running time analysis, the nested loop structure containing one “for” and one “while” loop, might be expressed as a pair of ________nested summations.

- $1$
- $2$
- $3$
- $4$

**Question #14**
The O-notation is used to state only the asymptotic ________ bounds.
Two
Lower
Upper (p25)
Both lower & upper

Question # 15
Efficient algorithm requires less computational…….
Memory
Running Time
Memory and Running Time
Energy

Question # 16
Consider the following Algorithm: Fun(n){ if (n=1) return 1 else return (n * Fun(n-1)) } Recurrence for the above
algorithm is:
nT(n-1)+1
2T(n-1)+1
T(n-1)+cn
T(n-1)+1

Question # 17
The time assumed for each basic operation to execute on RAM model of computation is-----
Infinite
Continuous
Constant (p10)
Variable

Question # 18
If the indices passed to merge sort algorithm are not equal, the algorithm may return immediately.
True False (p28)

Question # 19
Brute-force algorithm uses no intelligence in pruning out decisions.
True (p18) False

Question # 20
In analysis, the Lower Bound means the function grows asymptotically at least as fast as its largest term.
True False

Question # 21
In analysis, the Upper Bound means the function grows asymptotically no faster than its largest term.
True (p24) False

Question # 22
For small values of n, any algorithm is fast enough. Running time does become an issue when n gets large.
True (p14) Fast

Question # 23
The array to be sorted is not passed as argument to the merge sort algorithm.
Question # 24
In simple brute-force algorithm, we give no thought to efficiency.
True (p11) False

Question # 25
The ancient Roman politicians understood an important principle of good algorithm design that is plan-sweep algorithm.
Select correct option:
True
False (p27) [Divide and Conquer]

Question # 26
In 2d-space a point is said to be ________if it is not dominated by any other point in that space.
Select correct option:
Member
Minimal
Maximal (p11)
Joint

Question # 27
An algorithm is a mathematical entity that is dependent on a specific programming language.
Select correct option:
True
False (p7)

Question # 28
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.
True
False (p13)

Question # 29
8n^2 + 2n - 3 will eventually exceed \( c_2(n) \) no matter how large we make \( c_2(n) \).
True (p25)
False

Question # 30
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.
Fast
Slow
Expensive
Cheap (p11)

Question # 31
The running time of quick sort depends heavily on the selection of:
No of inputs
Arrangement of elements in array
Size of elements
Pivot element (p49)

Question # 32
Which sorting algorithm is faster?
O(n^2)
O(nlogn)
O(n+k) (p58)
O(n^3)

Question # 33
Q knapsack problem is called a “0-1” problem, because
Each item must be entirely accepted or rejected (p92)

Question # 34
Which statement is true?
Select correct option:
If a dynamic-programming problem satisfies the optimal-substructure property, then a locally optimal solution is globally optimal.
If a greedy choice property satisfies the optimal-substructure property, then a locally optimal solution is globally optimal.
Both of above (p77, p98)
None of above

Question # 35
Merge sort is stable sort, but not an in-place algorithm True (p54) False

Question # 36
In counting sort, once we know the ranks, we simply ______ numbers to their final positions in an output array.
Delete copy (p57) Mark arrange

Question # 37
Dynamic programming algorithms need to store the results of intermediate sub-problems. True (p75) False

Question # 38
A p × q matrix A can be multiplied with a q × r matrix B. The result will be a p × r matrix C. There are (p × r) total entries in C and each takes ______ to compute.
O (q) O (1) (p84) O (n^2) O (n^3)

Question # 39
Due to left-complete nature of binary tree, heaps can be stored in
Link list Structure Array (p40)
None of above

Question # 40
Which of the following is calculated with big O notation?
 Lower bounds  **Upper bounds**
 Both upper and lower bound  Medium bounds

  - The definition of Θ-notation relies on proving both a lower and upper asymptotic bound.
  - The O-notation is used to state only the asymptotic upper bounds.

**Question # 41**
Merge sort makes two recursive calls. Which statement is true after these recursive calls finish, but before the merge step?
The array elements form a heap
**Elements in each half of the array are sorted amongst themselves**
Elements in the first half of the array are less than or equal to elements in the second half of the array
None of the above

**Question # 42**
Who invented Quick sort procedure?
Hoare  Sedgewick  Mellroy  Coreman

**Question # 43**
Merge sort requires extra array storage.  **True (p54)**  False
  - Mergesort is a stable algorithm but **not** an in-place algorithm. It requires extra array storage.

**Question # 44**
In which order we can sort?
Increasing order only  decreasing order only
**Increasing order or decreasing order**  both at the same time

**Question # 45**
In the analysis of Selection algorithm, we make a number of passes, in fact it could be as many as, 

  \[ T(n) \quad T(n / 2) \quad \log n \quad (p37) \quad n / 2 + n / 4 \]

**Question # 46**
How much time merge sort takes for an array of numbers?

  \[ T(n^2) \quad T(n) \quad T(\log n) \quad T(n \log n) \quad (p40) \]

**Question # 47**
Sieve Technique applies to problems where we are interested in finding a single item from a larger set of ___

  \[ n \text{ items (p34)} \quad \text{phases} \quad \text{pointers} \quad \text{constant} \]

**Question # 48**
The sieve technique works in __________ as follows

  \[ \text{Phases (p34)} \quad \text{numbers} \quad \text{integers} \quad \text{routines} \]

**Question # 49**
For the heap sort, access to nodes involves simple __________ operations.

  \[ \text{Arithmetic (p41)} \quad \text{binary} \quad \text{algebraic} \quad \text{logarithmic} \]

**Question # 50**
The analysis of Selection algorithm shows the total running time is indeed ________ in n,
Question # 51
Slow sorting algorithms run in,
\[ T(n^2) \quad (p39) \quad T(n) \quad T(\log n) \quad T(n\log n) \]

Question # 52
A heap is a left-complete binary tree that conforms to the
Increasing order only decreasing order only \textit{heap order} \quad (\log n) order

Question # 53
For the heap sort we store the tree nodes in
\textit{Level-order traversal} \quad (p40) \quad \textit{in-order traversal} \quad \textit{pre-order traversal} \quad \textit{post-order traversal}

Question # 54
In Sieve Technique we do not know which item is of interest \quad \textbf{True} \quad (p34) \quad \textbf{False}

Question # 55
Divide-and-conquer as breaking the problem into a small number of
Pivot \quad Sieve \quad \textit{smaller sub problems} \quad (p34) \quad Selection

Question # 56
For the sieve technique we solve the problem,
\textit{Recursively} \quad (p34) \quad \textit{mathematically} \quad \textit{precisely} \quad \textit{accurately}

Question # 57
The reason for introducing \textit{Sieve Technique} algorithm is that it illustrates a very important special case of,
\textit{Divide-and-conquer} \quad (p34) \quad \textit{decrease and conquer} \quad \textit{greedy nature} \quad 2\text{-dimension Maxima}

Question # 58
Sorting is one of the few problems where provable _____ bonds exits on how fast we can sort,
Upper \quad \textit{Lower} \quad (p39) \quad \textit{Average} \quad \log n

Question # 59
Sieve Technique can be applied to selection problem? \quad \textbf{True} \quad (p35) \quad \textbf{False}

Question # 60
One of the clever aspects of heaps is that they can be stored in arrays without using any______
\textit{Pointers} \quad (p40) \quad \textit{constants} \quad \textit{variables} \quad \textit{functions}

Question # 61
Heaps can be stored in arrays without using any pointers; this is due to the ____ nature of the binary tree,
\textit{Left-complete} \quad (p40) \quad \textit{right-complete} \quad \textit{tree nodes} \quad \textit{tree leaves}

Question # 62
How many elements do we eliminate in each time for the Analysis of Selection algorithm?
\textit{n / 2 elements} \quad (p36) \quad (n / 2) + n elements \quad n / 4 elements \quad 2 n elements

Question # 63
The sieve technique is a special case, where the number of sub problems is just
Question # 64
For the Sieve Technique we take time
\[ T(nk) \quad T(n/3) \quad n^2 \quad n/3 \]

Question # 65
The number of nodes in a complete binary tree of height \( h \) is
Select correct option:
1. \( 2^{(h+1)} - 1 \)
2. \( 2 \times (h+1) - 1 \)
3. \( 2 \times (h+1) \)
4. \( ((h+1)^2) - 1 \)

Question # 66
Random access machine or RAM is a/an
Machine build by Al-Khwarizmi
Mechanical machine
Electronics machine
Mathematical model (p10)

Question # 67
Analysis of Selection algorithm ends up with,
\[ T(n) \]
\[ T(1/1 + n) \]
\[ T(n/2) \]
\[ T((n/2) + n) \]

Question # 68
Continuation sort is suitable to sort the elements in range 1 to \( k \)
1. \( K \) is Large 2. \( K \) is not known 3. \( K \) may be small or large 4. \( K \) is small (p57)

Question # 69
Counting sort is suitable to sort the elements in range 1 to \( k \):
K is large   K is small   K may be large or small   None

Question # 70
Continuing sort has time complexity of?
1. \( O(n) \) (p58) 2. \( O(n+k) \)   3. \( O(n\log n) \)

Question # 71
Counting sort has time complexity:
\( O(n) \) (p58) \( O(n+k) \) \( O(k) \) \( O(n\log n) \)

Question # 72
Memoization is:
To store previous results for further use.
To avoid unnecessary repetitions by writing down the results of recursive calls and looking them again if needed later. (p74)
To make the process accurate
None of the above

**Question # 73**
In Quick sort algorithm, constants hidden in T(n lg n) are:
Large       Medium       Not known       **small**       (Ref)

**Question # 74**
Quick sort is based on divide and conquer paradigm; we divide the problem on base of pivot element and:
**There is explicit combine process as well to conquer the solution.**
No work is needed to combine the sub-arrays, the array is already sorted
Merging the subarrays
None of above.

**Question # 75**
In RAM model instructions are executed
**One after another**       (p10)
Parallel
Concurrent
Random

**Question # 76**
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,
linear       arithmetic       **geometric**       (p37)       exponent

**Question # 77**
**________** is a graphical representation of an algorithm
Σ notation       Θ notation       **Flowchart**       (Ref)       Asymptotic notation

**Question # 78**
A RAM is an idealized machine with _____ random-access memory.
256MB       512MB       An infinitely large       (p10)       100GB

**Question # 79**
What type of instructions Random Access Machine (RAM) can execute? Choose best answer
Algebraic and logic
Geometric and arithmetic
**Arithmetic and logic**       (p10)
Parallel and recursive

**Question # 80**
What will be the total number of **max comparisons** if we run **brute-force maxima** algorithm with n elements?
\[ n^2 \]       (p14)       \[ 2n/n \]       \[ n \]       \[ 8n \]

**Question # 81**
What is the solution to the recurrence T(n) = T(n/2)+n .
O(logn)       **O(n)**       O(nlogn)       O(n2)

**Question # 82**
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(); }
```

**Question # 83**
What is the order of execution for this code.
- $O(n)$
- $O(n^3)$
- $O(n^2 \log n)$
- $O(n^2)$

**Question # 84**
Consider the following Algorithm:
```
Factorial (n){
    if (n=1)
        return 1
    else
        return (n * Factorial(n-1))
}
```

Recurrence for the following algorithm is:
- $T(n) = T(n-1) + 1$
- $T(n) = nT(n-1) + 1$
- $T(n) = T(n-1) + n$
- $T(n) = T(n(n-1)) + 1$

**Question # 85**
What is the total time to heapify?
- $O(\log n)$  (p43)
- $O(n \log n)$
- $O(n^2 \log n)$
- $O(\log^2 n)$

**Question # 86**
When we call heapify then at each level the comparison performed takes time
- $\Theta(1)$  (p43)
- Time will vary according to the nature of input data
- It cannot be predicted
- It will take $\Theta \log n$

**Question # 87**
In Quick sort, we don’t have the control over the sizes of recursive calls
- True  (p49)
- False
- Less information to decide
- Either true or false

**Question # 88**
Is it possible to sort without making comparisons?  Yes  (p57)
- No

**Question # 89**
If there are $\Theta(n^2)$ entries in edit distance matrix then the total running time is
- $\Theta(1)$
- $\Theta(n^2)$
Question # 90
For Chain Matrix Multiplication we cannot use divide and conquer approach because,

We do not know the optimum \(k\) \textcolor{green}{(p86)}
- We use divide and conquer for sorting only
- We can easily perform it in linear time
- Size of data is not given

Question # 91
The Knapsack problem belongs to the domain of ______________ problems.

Optimization \textcolor{green}{(p91)}   NP Complete   Linear Solution   Sorting

Question # 92
Suppose we have three items as shown in the following table, and suppose the capacity of the knapsack is 50 i.e. \(W = 50\).

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>60</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>30</td>
</tr>
</tbody>
</table>

The optimal solution is to pick
- Items 1 and 2
- Items 1 and 3
- Items 2 and 3
- None of these

Question # 93
Theta asymptotic notation for \(T(n)\):
Select correct option:
- Set of functions described by: \(c_1g(n) \leq f(n)\) for \(c_1\) some constant and \(n=n_0\)
- Set of functions described by \(c_1g(n) \geq f(n)\) for \(c_1\) some constant and \(n=n_0\)
- Theta for \(T(n)\) is actually upper and worst case complexity of the code
- Set of functions described by: \(c_1g(n) \leq f(n) \leq c_2g(n)\) for \(c_1\) and \(c_2\) some constants and \(n=n_0\)

Question # 94
A (an) _____ is a left-complete binary tree that conforms to the heap order.

Heap \textcolor{green}{(page 40)}
- Binary tree
- Binary search tree
- Array

Question # 95
A heap is a left-complete binary tree that conforms to the __________ order
- increasing order only
- decreasing order only
- heap order \textcolor{green}{(p40)}
- \((\log n)\) order
Question # 96
Consider the following Algorithm: Fun(n){ if (n=1) return 1 else return (n * Fun(n-1)) } Recurrence for the above algorithm is: Select correct option:

- nT(n-1)+1
- 2T(n-1)+1
- T(n-1)+cn
- T(n-1)+1

Question # 97
The recurrence relation of Tower of Hanoi is given below T(n)={ 1 if n=1 and 2T(n-1) if n >1 In order to move a tower of 5 rings from one peg to another, how many ring moves are required? Select correct option: 16 10 32 (p30) 31

Question # 98
In Quick Sort Constants hidden in T(n log n) are
1. Large
2. Medium
3. Small (Reference)
4. Not Known

Question # 99
In stable sorting algorithm:
One array is used
In which duplicating elements are not handled.
More than one arrays are required.
**Duplicating elements remain in same relative position after sorting.** (p54)

Question # 100
In in-place sorting algorithm is one that uses arrays for storage:
An additional array
**No additional array** (p54)
Both of above may be true according to algorithm
More than 3 arrays of one dimension

Question # 101
Which may be a stable sort?
1. Merger
2. Insertion
3. Both above (p54)
4. None of the above

Question # 102
An in place sorting algorithm is one that uses ___ arrays for storage
1. Two dimensional arrays
2. More than one array
3. No Additional Array (p54)
4. None of the above

Question # 103
Quick sort is
Stable and In place
**Not stable but in place** (p54)
Stable and not in place
Sometime in place and sometime stable

Question # 104
One Example of in place but not stable sort is Quick (p54) Heap Merge Bubble

**Question # 105**
Which may be stable sort?
- Bubble sort Insertion sort both of above (p54)

**Question # 106**
Merge sort is stable sort, but not an in-place algorithm True (p54) False

An in-place sorting algorithm is one that uses no additional array for storage. A sorting algorithm is stable if duplicate elements remain in the same relative position after sorting. (p54)

<table>
<thead>
<tr>
<th>9</th>
<th>3</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>5</th>
<th>2</th>
<th>1</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>

Bubble sort, insertion sort and selection sort are in-place sorting algorithms.

Bubble sort and insertion sort can be implemented as stable algorithms but selection sort cannot (without significant modifications).

Mergesort is a stable algorithm but not an in-place algorithm. It requires extra array storage.

Quicksort is not stable but is an in-place algorithm.

Heapsort is an in-place algorithm but is not stable.

---

**Final Term MCQS**

**Question # 14**
A dense undirected graph is:
Select correct option:
- A graph in which \( E = O \left(V^2\right) \) (Reference)
- A graph in which \( E = O \left(V\right) \)
- A graph in which \( E = O \left(\log V\right) \)
- All items above may be used to characterize a dense undirected graph

**Question # 15**
What algorithm technique is used in the implementation of Kruskal solution for the MST?
- Greedy Technique (p142)
- Divide-and-Conquer Technique
- Dynamic Programming Technique
The algorithm combines more than one of the above techniques

**Question # 16**
A digraph is strongly connected under what condition?
A digraph is strongly connected if for every pair of vertices \( u, v \in V \), \( u \) can reach \( v \).
**A digraph is strongly connected if for every pair of vertices \( u, v \in V \), \( u \) can reach \( v \) and vice versa.** (p135)
A digraph is strongly connected if for at least one pair of vertex \( u, v \in V \), \( u \) can reach \( v \) and vice versa.
A digraph is strongly connected if at least one third pair of vertices \( u, v \in V \), \( u \) can reach \( v \) and vice versa.

**Question # 17**
The relationship between number of back edges and number of cycles in DFS is,
Both are equal
Back edges are half of cycles
Back edges are one quarter of cycles
**There is no relationship between no. of edges and cycles** (p131)

**Question # 18**
Suppose that a graph \( G = (V,E) \) is implemented using adjacency lists. What is the complexity of a breadth-first traversal of \( G \)? Select correct option:
- \( O(|V|^2) \)
- \( O(|V| |E|) \)
- \( O(|V|^2 |E|) \)
- \( O(|V| + |E|) \) (p116)

**Question # 19**
**Forward edge is**?
Select correct option:
- \( (u, v) \) where \( u \) is a proper descendent of \( v \) in the tree.
- **(u, v) where \( v \) is a proper descendent of \( u \) in the tree.** (p129)
- \( (u, v) \) where \( v \) is a proper ancestor of \( u \) in the tree.
- \( (u, v) \) where \( u \) is a proper ancestor of \( v \) in the tree.

**Question # 20**
**Back edge is**?
Select correct option:
- \( (u, v) \) where \( v \) is an ancestor of \( u \) in the tree. (p128)
- \( (u, v) \) where \( u \) is an ancestor of \( v \) in the tree.
- \( (u, v) \) where \( v \) is a predecessor of \( u \) in the tree.
- None of above

**Question # 21**
**Cross edge is**?
- \( (u, v) \) where \( u \) and \( v \) are not ancestor of one another
- \( (u, v) \) where \( u \) is ancestor of \( v \) and \( v \) is not descendent of \( u \).
- **(u, v) where \( u \) and \( v \) are not ancestor or descendent of one another** (p129)
- \( (u, v) \) where \( u \) and \( v \) are either ancestor or descendent of one another.

**Question # 22**
If you find yourself in maze the better traversal approach will be?
Select correct option:
BFS and DFS both are valid (p119)

BFS

DFS

Question # 23
In digraph G=(V,E) ;G has cycle if and only if
Select correct option:
The DFS forest has forward edge.
The DFS forest has back edge (p131)
The DFS forest has both back and forward edge
BFS forest has forward edge

Question # 24
What is generally true of Adjacency List and Adjacency Matrix representations of graphs?
Select correct option:
Lists require less space than matrices but take longer to find the weight of an edge (v1,v2)
Lists require less space than matrices and they are faster to find the weight of an edge (v1, v2) (p116)
Lists require more space than matrices and they take longer to find the weight of an edge (v1, v2)
Lists require more space than matrices but are faster to find the weight of an edge (v1, v2)

Question # 5
There is relationship between number of back edges and number of cycles in DFS
Select correct option:
Both are equal.
Cycles are half of back edges.
Cycles are one fourth of back edges.
There is no relationship between back edges and number of cycle (p131)

Question # 6
You have an adjacency list for G, what is the time complexity to compute Graph transpose G^T.? Select correct option:
(V+E) (Reference)
V.E
V
E

Question # 7
Dijkstra’s algorithm:
Has greedy approach to find all shortest paths
Has both greedy and dynamic approach to find all shortest paths
Has greedy approach to compute single source shortest paths to all other vertices (p154)
Has both greedy and dynamic approach to compute single source shortest paths to all other vertices.

Question No: 25
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 vertices.
True False
Question No: 26
If a problem is in NP, it must also be in P.
   True   False (p178)   unknown

Question No: 27
If a problem is in NP-complete, it must also be in NP.
   True   False

Question No: 28
Maximum number of vertices in a Directed Graph may be $|V^2|$
   True   False

Question No: 29
The Huffman algorithm finds a (n) ___________ solution.
   Optimal (Reference)   Non-optimal   Exponential   Polynomial

Question No: 30
The Huffman algorithm finds an exponential solution   True   False (Reference)

Question No: 31
The Huffman algorithm finds a polynomial solution   True   False (Reference)

Question No: 32
The codeword assigned to characters by the Huffman algorithm have the property that no codeword is the postfix of any other.   True   False

Question No: 33
The codeword assigned to characters by the Huffman algorithm have the property that no codeword is the prefix of any other.   True (p101)   False

Prefix Property:
The codewords assigned to characters by the Huffman algorithm have the property that no codeword is a prefix of any other:

Question No: 48
Non-optimal or greedy algorithm for money change takes ___________
   O(k) (p99)   O(kN)   O(2k)   O(N)

Question No: 49
The Huffman codes provide a method of encoding data inefficiently when coded using ASCII standard.
   True   False (p99)
   o The Huffman codes provide a method of encoding data efficiently.

Question No: 50
Using ASCII standard the string “abacdaacacwe” will be encoded with __________ bits
Select correct option:
   64
   128
   96 (p101 12x8=96)
Question No: 51
Using ASCII standard the string abacdaacac will be encoded with __________ bits.

- Consider the string “ abacdaacac”. if the string is coded with ASCII codes, the message length would be 10 × 8 = 80 bits.

Question No: 52
Using ASCII standard the string abacdaacac will be encoded with 160 bits. True False (p99)

Question No: 53
Using ASCII standard the string abacdaacac will be encoded with 320 bits. True False (p99)

Question No: 54
Using ASCII standard the string abacdaacac will be encoded with 100 bits. True False (p99)

Question No: 55
Using ASCII standard the string abacdaacac will be encoded with 32 bytes. True False (p99)

Question No: 56
The greedy part of the Huffman encoding algorithm is to first find two nodes with smallest frequency. True (p100) False

Question No: 57
The greedy part of the Huffman encoding algorithm is to first find two nodes with character frequency. True False (p100)

Question No: 58
The greedy part of the Huffman encoding algorithm is to first find two nodes with larger frequency. True False

Question No: 59
Dijkstra’s single source shortest path algorithm works if all edges weights are non-negative and there are negative cost cycles. True False

Question No: 60
Dijkstra’s single source shortest path algorithm works if all edges weights are non-negative and there are no negative cost cycles. True (p159) False

Question No: 61
Dijkstra’s single source shortest path algorithm works if all edges weights are negative and there are no negative cost cycles. True False

Question No: 62
Floyd-Warshall algorithm is a dynamic programming algorithm; the genius of the algorithm is in the clever recursive formulation of the shortest path problem. True (p162) False

Question No: 63
Floyd-Warshall algorithm, as in the case with DP algorithms, we avoid recursive evaluation by generating a table for $d_{ij}(k)$

True (p164)

False

**Question No: 64**
The term coloring came from the original application which was in map drawing.   True (p173)   False

The term “coloring” came from the original application which was in architectural design. True False

**Question No: 65**
In the clique cover problem, for two vertices to be in the same group, they must be _____ each other.
Apart from Far from Near to Adjacent to (p176)

**Question No: 66**
In the clique cover problem, for two vertices to be in the same group, they must be adjacent to each other.  True False

**Question No: 67**
In the clique cover problem, for two vertices to be in the same group, they must be apart from each other.  True False (p176)

**Question No: 68**
The difference between Prims algorithm and Dijkstra s algorithm is that Dijkstra s algorithm uses a different key.  True (p156) False

**Question No: 69**
The difference between Prim s algorithm and Dijkstra s algorithm is that Dijkstra s algorithm uses a same key.  True False (p156)

**Question No: 70**
You have an adjective list for G, what is the time complexity to computer graph transpose $G^T$?

$(V + E)$   (p138)

**Question No: 71**
Given an adjacency list for G, it is possible to compute $G^T$ in $\Theta(V + E)$ time.
It takes $O(\log V)$ to extract a vertex from the priority queue.

**Question No: 72**
Overall time for Kruskal is $\Theta(E \log E) = \Theta(E \log V)$ if the graph is sparse.  (p149) True

**Question No: 73**
An optimization problem is one in which you want to find,
Not a solution
An algorithm
Good solution
**The best solution**

**Question No: 74**
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 vertices.

True
False

Question No: 75
If a graph has \( v \) vertices and \( e \) edges then to obtain a spanning tree we have to delete
- \( v \) edges.
- \( v - e + 5 \) edges
- \( v + e \) edges.

None of these

Question No: 76
Huffman algorithm uses a greedy approach to generate an antefix code \( T \) that minimizes the expected length \( B(T) \) of the encoded string.  

True  False  (p102)

Question No: 77
Huffman algorithm uses a greedy approach to generate a postfix code \( T \) that minimizes the expected length \( B(T) \) of the encoded string.  

True  False  (p102)

Question No: 78
Huffman algorithm uses a greedy approach to generate a prefix code \( T \) that minimizes the expected length \( B(T) \) of the encoded string.  

True  (p102)  False

Question No: 79
Bellman-Ford allows negative weights edges and negative cost cycles.
- True  False  (p159)
- Bellman-Ford allows negative weights edges and no negative cost cycles.

Question No: 80
We do sorting to,
- keep elements in random positions
- keep the algorithm run in linear order
- keep the algorithm run in \((\log n)\) order
- keep elements in increasing or decreasing order

Question No: 81
After partitioning array in Quick sort, pivot is placed in a position such that

Values smaller than pivot are on left and larger than pivot are on right  (Ref)

Values larger than pivot are on left and smaller than pivot are on right
Pivot is the first element of array
Pivot is the last element of array

Question No: 82
Dynamic programming algorithms need to store the results of intermediate sub-problems. True  False

Question No: 83
Which statement is true?
If a dynamic-programming problem satisfies the optimal-substructure property, then a locally optimal solution is globally optimal.
If a greedy choice property satisfies the optimal-substructure property, then a locally optimal solution is globally optimal.

Both of above  None of above

Question No: 84
What general property of the list indicates that the graph has an isolated vertex?
There is null pointer at the end of list.
The isolated vertex is not handled in list.
Only one value is entered in the list.
There is at least one null list.

Question No: 85
Depth first search is shortest path algorithm that works on un-weighted graphs. True False (p153)
   o The breadth-first-search algorithm is a shortest-path algorithm that works on un-weighted graphs.

Question No: 86
Which is true statement?
Select correct option:
Breadth first search is shortest path algorithm that works on un-weighted graphs (p153)
Depth first search is shortest path algorithm that works on un-weighted graphs.
Both of above are true.
None of above are true.

Question # 9
Which is true statement in the following?
Kruskal algorithm is multiple source technique for finding MST.
Kruskal’s algorithm is used to find minimum spanning tree of a graph, time complexity of this algorithm is O(EV)
Both of above
Kruskal's algorithm (choose best non-cycle edge) is better than Prim's (choose best tree edge) when the graph has relatively few edges.

Question # 11
Kruskal's algorithm (choose best non-cycle edge) is better than Prim's (choose best tree edge) when the graph has relatively few edges.
True False

Question # 8
What is the time complexity to extract a vertex from the priority queue in Prim’s algorithm?
Select correct option:
O (log E)
(V)
(V+E)
O (log V) (p152)

(1) In Prim’s algorithm, the additional information maintained by the algorithm is the length of the shortest edge from vertex v to points already in the tree.
a) True b) False c) unknown
(2) 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 vertices**.

a) True  
b) False  
c) unknown

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

a) True  
b) False  
c) unknown

(4) Which statement is true?

(i) The running time of Bellman-Ford algorithm is $T(VE)$
(ii) Both Dijkstra’s algorithm and Bellman-Ford are based on performing **repeated relaxations**
(iii) The 0-1 knapsack problem is hard to solve

Only i  
Only iii  
Both i and iii  
**All of these**

5) Which of the following arrays represent **descending (max) heaps**?

i. [10,7,7,2,4,6]  
ii. [10,7,6,2,4,7]  
iii. [10,6,7,2,4,6]  
iv. [6,6,7,2,4,10]  

Only ii  
Only iv  
Both ii and iv  
Both i and iii

6. Which of the following statement(s) is/are correct?

(a) $O(n \log n + n^2) = O(n^2)$.  
(b) $O(n \log n + n^2) = O(n^2 \log 2n)$  
(c) $O(c n^2) = O(n^2)$ where $c$ is a constant.  
(d) $O(c n^2) = O(c)$ where $c$ is a constant.  
(e) $O(c) = O(1)$ where $c$ is a constant.

**Only (a) & (e)**  
Both (c) and (e)

7. Which of the shortest path algorithms would be most appropriate for finding paths in the graph with **negative edge weights and cycles**?

i. Dijkstra’s Algorithm  
ii. Bellman-Ford Algorithm  
iii. Floyd Warshall Algorithm  

Only ii  
Only iii  
Both ii & iii

9. Suppose we have two problems A and B. Problem A is polynomial-time reducible and problem B is NP-complete. If we reduce problem A into B then problem A becomes NP-complete  
Yes  
No

12. Edge $(u, v)$ is a forward edge if

u is a proper descendant of v in the tree  
$v$ is a proper descendant of $u$ in the tree  

None of these

14. If, in a DFS forest of digraph $G = (V, E)$, $f[u] = f[v]$ for an edge $(u, v) \in E$ then the edge is called **Back edge**  
Forward edge  
Cross Edge  
Tree Edge  
None of these

16. Best and worst case times of an algorithm may be same.  
**True**  
False

17. Can an adjacency matrix for a directed graph ever not be square in shape?  
Yes  
No

**Question No: 44**

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

10 00 010
Question No: 45
Can an adjacency matrix for a directed graph ever not be square in shape? Yes No
  o No. since we want to describe the relationship between each node and each other node, we need precisely n^2 matrix entries.

Question No: 34
Shortest path problems can be solved efficiently by modeling the road map as a graph. True False

Question No: 35
Dijkstra’s algorithm is operates by maintaining a subset of vertices. True False