*Placement handbook of*

# ALGORITHMS

## TOP 40 MCQ'S WITH ANSWERS

# TOP 40 ALGORITHM MCQ's WITH ANSWERS

Q1. Consider a truck that shifts products from source to destination. There are N products of different weights. The truck driver can make one trip in a day and can take a maximum load of X with him. Which algorithm can be used in order to maximize the number of products he can take in one trip?

A. Back tracking
B. Greedy approach
C. Dynamic programming
D. Brute force

**Answer B**

Explanation:
The above problem can be solved by the algorithm i.e. Container loading problem which is a type of greedy approach.

---

Q2.What is the space complexity of following code snippet

```
int sum(int A[], int n)
{
        int sum = 0, i;
        for(i = 0; i< n; i++)
        sum = sum + A[i];
        return sum;
}
```

A. n+8
B. 2n+16
C. 2n+8
D. n^2

**Answer C**

Explanation:
'**n*2**' bytes of memory to store array variable '**a[]**'. 2 bytes of memory for integer parameter '**n**'
4 bytes of memory for local integer variables '**sum**' and '**i**' (2 bytes each) 2 bytes of memory

for **return value**. That means, totally it requires **'2n+8'** bytes of memory to complete its execution. Here, the amount of memory depends on the input value of **'n'.**

---

Q3. What is the time complexity of the given code snippet? (Choose the most appropriate)
A. O(n^2)
B. O(n)
C. O(3n+2)
D. 0(n^3)

```
for(i=0;i<n;i++)
{
        sum=sum+a[i];
}
```

**Answer B**

Explanation

| Statement | Frequency Count |
|---|---|
| i=0 | 1 |
| i<n | (n+1) |
| i++ | n |
| sum=sum+a[i] | n |
| **Total** | **3n+2** |

**Frequency Count:** 3n+2
Time Complexity is **O(n)** (**Note**: Ignore the constants value)

---

Q4. What is the time complexity of n nested for loops present in any code?
A. O(n^n)
B. O(n)
C. O(nlogn)
D. 0(n^3)

**Answer A**

Explanation
For any number of loops present in nested form, the time complexity is n raise to the power of maximum no of nested loops present in a program.

---

Q5 Which of the following is true about Huffman Coding.

A. Huffman coding may become lossy in some cases
B. Huffman Codes may not be optimal lossless codes in some cases
C. In Huffman coding, no code is prefix of any other code.
D. All of the above

**Answer C**

Huffman coding is a lossless data compression algorithm. The codes assigned to input character are prefix codes, means the codes are assigned in such a way that the code assigned to one character is not prefix of code assigned to any other character. This is how Huffman Coding makes sure that there is no ambiguity when decoding.

---

Q6. Suppose you are provided with a document of 1000 lines and a string pattern whose presence is to be checked in the entire document. Which algorithm strategy is reliable to provide 100 % accurate result?

A. Backtracking
B. Brute force
C. Huff man coding
D. Activity selection

**Answer B**

Explanation
Brute force technique guarantees 100 % accurate results. The technique is however avoided as it is not an efficient algorithm

---

Q7 What is the worst case of linear search?

A. O(n)
B. O(1)
C. O(n^2)
D. O(logn)

**Answer A**

Explanation:
The worst case scenario can be that the element is present at the end or not present. In both situations we have to iterate through the entire array. If the size of the array in n then we will have N checks, so time complexity is n.

---

Q8 Which algorithm strategy can be applied in order to find the most optimal solution between two points?

A. Dijkstra's algorithm
B. Multi stage graph
C. Floyd-War shall algorithm
D. Knapsack

Ans. B
Explanation:
Multi multi-stage graph is a dynamic algorithm which can be used as a greedy approach does not guarantee an optimal solution.

---

Q9 What is the recurrence expression for Strassen's matrix multiplication?

A. T(n) = 8T(n/2) + n^2
B. T(n) = 7T(n/2) + n^2
C. T(n) = 8T(n/2) + n^3
D. T(n) = 8T(n) + n2

**Answer B**
Explanation

Strassen's reduces the number of recursion call from 8 to 7 using its own formula which results in decrease of time complexity.

---

Q10 Which of the following algorithm implementations is similar to that of an insertion sort?

| |
|---|
| A. Binary heap |
| B. Quick sort |
| C. Merge sort |
| D. Radix sort |

**Answer A**

Explanation:

Insertion sort is similar to that of a binary heap algorithm because of the use of temporary variable to swap.

---

Q11.A person wants to visit some places. He starts from a vertex and then wants to visit every vertex till it finishes from one vertex, backtracks and then explore other vertex from same vertex. What algorithm he should use?

A. Depth first search
B. Breath first search
C. Trim's Algorithm
D. Prim's Algorithm

**Answer A**

Explanation

This is the definition of the Depth First Search. Exploring a node, then aggressively finding nodes till it is not able to find any node.

---

Q12. You are given infinite coins of denominations 5, 7, 9. Which of the following sum CANNOT be achieved using these coins?

a) 50

b) 21

c) 13

d) 23

**Answer C**

The given problem can be solved using greedy approach. The scenario given in the question refers to the famous coin exchange problem.
Explanation: One way to achieve a sum of 50 is to use ten coins of 5. A sum of 21 can be achieved by using three coins of 7. One way to achieve a sum of 23 is to use two coins of 7 and one coin of 9. A sum of 13 cannot be achieved.

---

Q13 How many passes are required to sort a file of size N by bubble sort

A. N^2
B. N
C. N-1
D. N/2

**Answer C**
Explanation:
The number of passes in bubble sort is one less than the size of the file or input elements.

---

Q 14 Which of the following sorting procedures is the slowest?

A. Quick sort
B. Heap sort
C. Shell sort
D. Bubble sort

**Answer D**

Explanation
Bubble sort mechanism works on swaps and we time complexity of swaps is O(n^2) thus the technique is slowest.

---

Q15. Consider a situation where swap operation is very costly. Which of the following sorting algorithms should be preferred so that the number of swap operations are minimized in general?

A. Insertion sort
B. Selection sort
C. Heap sort
D. Merge sort

**Answer B**

Explanation

Selection sort makes O(n) swaps which is minimum among all sorting algorithms mentioned above.

---

Q16_____ enumerates a list of promising nodes that could be computed to give the possible solutions of a given problem.

A. Exhaustive search
B. Brute Force
C. Back tracking
D. Divide and conquer

**Answer C**

Explanation
Backtracking is a general algorithm that evaluates partially constructed candidates that can be developed further without violating problem constraints.

---

Q17. What is the time complexity of Kruskal's algorithm?

| | |
|---|---|
| A | O(log V) |
| B | O(E log V) |
| C | $O(E^2)$ |
| D | O(V log E) |

**Answer B**

Explanation

Kruskal's algorithm involves sorting of the edges, which takes O (E logE) time, where E is a number of edges in graph and V is the number of vertices. After sorting, all edges are iterated and union-find algorithm is applied. union-find algorithm requires O(logV) time. So, overall Kruskal's algorithm requires O (E log V) time.

Q18. Match the following?

| | |
|---|---|
| 1. Crossword | A. Dynamic programming |
| 2. Fractional Knapsack | B. Brute force |
| 3. Binary Search | C. Backtracking |
| 4. Principle of optimality | D. Greedy approach |

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

**Answer C**
Explanation:

- Crossword puzzles are based on backtracking approach as in crossword you move ahead try a solution if it fails then come back to previous and try another option
- Greedy algorithm is used to solve fractional knapsack problem. We first sort items according to their value/weight ratio and then add item with highest ratio until we cannot add the next item as a whole. At the end, we add the next item as much as we can.
- Binary Search is implemented by using Brute Force technique
- Dynamic programming works on the principle of optimality

---

Q19 Imagine that you have 5 friends: Billy, Jenna, Cassie, Alyssa, and Harry. You know a few roads that connect some of their houses, and you know the lengths of those roads. Which algorithm you will use to help all of these connect to each other in the most optimal way.

A. Floyd war shall
B. Dijkstra's algorithm
C. Multistage graph
D. Huffman Tree

**Answer A**

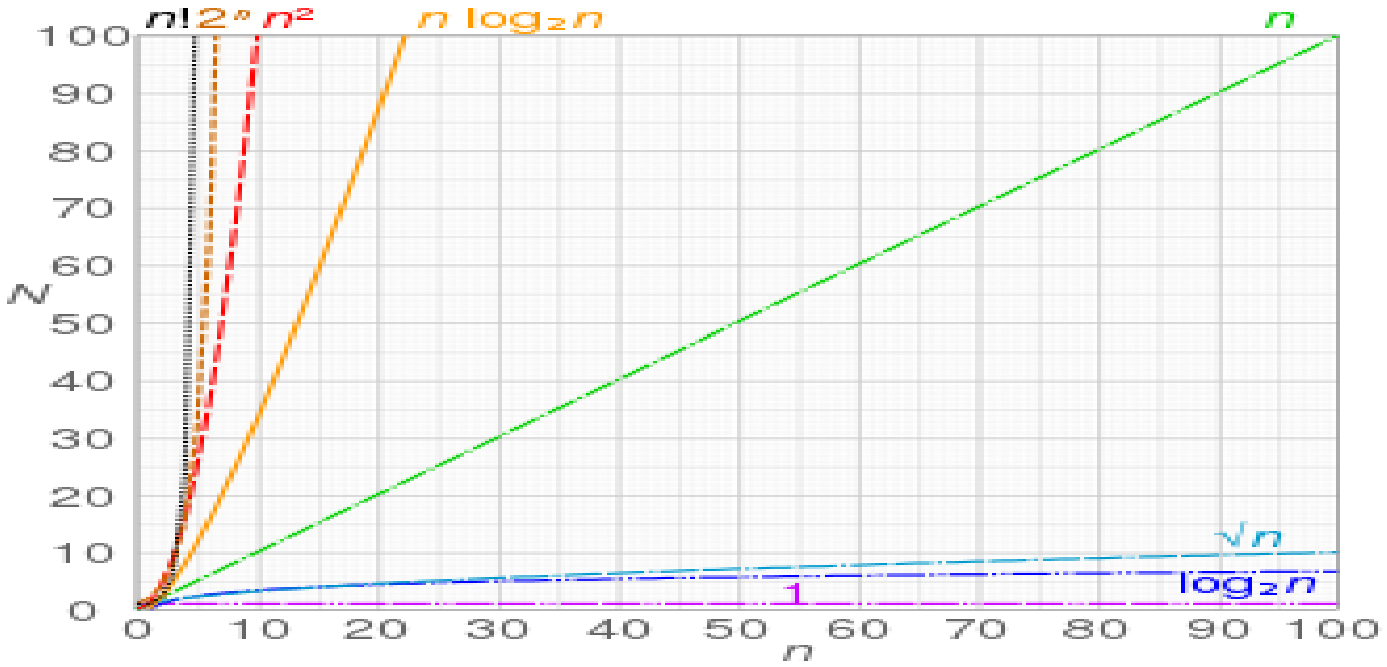
Explanation:
Floyd Warshall algorithm provides shortest distance between all pair of points thus provides an optimal solution.

**Q 20** Arrange the below time complexity in order of the most to least efficient?

1. NlogN
2. N
3. N^2
4. N!

A. 2,1,3,4
B. 1,2,3,4
C. 4,2,3,1
D. 3,2,1,4

Answer A
Explanation:




Q 21. If for an algorithm time complexity is given by O ((3/2) ^n) then complexity will:

A. constant
B. quadratic
C. exponential
D. none of the mentioned

**Answer C**
Explanation: The growth rate of that function will be exponential therefore complexity will be exponential

---

Q22 Given an array arr = {5,6,77,88,99} and key = 88; How many iterations are done until the element is found in a binary search?

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

**Answer A**

Explanation: Iteration1: mid = 77; Iteration2: mid = 88;

---

Q23. Which of the following is not the property of algorithm?

A. Finiteness
B. Definiteness
C. Sequence
D. Order of growth

**Answer D**
Explanation
Measuring the performance of an algorithm in relation with the input size n is called order of growth.

---

Q24. Which is not a primitive data type?

A. Pointers
B. Files
C. Integer
D. String constant

**Answer B**

Explanation
File is a type of non primitive data type.

---

Q25 Arrange the steps in order with reference to the steps for designing a algorithm

1. Decision making on choice of algorithm, strategies
2. Decision making on choice of data structure
3. Analysis of algorithm
4. Verification of algorithm

A. 1,2,3,4
B. 2,1,3,4
C. 2,1,4,3
D. 1,3,4,2

**Answer C**

---

Q26 _____ is the amount of memory used to store compiled version of instruction

A. Instruction space
B. Environmental stack
C. Data space
D. String pool

**Answer A**

Explanation:

Instruction Space: It is the amount of memory used to store compiled version of instructions.
Environmental Stack: It is the amount of memory used to store information of partially executed functions at the time of function call.
Data Space: It is the amount of memory used to store all the variables and constants
String pool: Present inside heap to store string literals.

---

Q27. Which of the following is an NP complete problem?

A. Hamiltonian cycle
B. Travelling salesman problem
C. Calculating chromatic number of graph
D. Finding maximum element in an array

**Answer C**

Explanation:

- Calculating the chromatic number of a graph is an NP complete problem as a chromatic number of an arbitrary graph cannot be determined by using any convenient method.
- Hamiltonian cycle or circuit is a Hamiltonian path, that there is an edge from the last vertex to the first vertex. These type of problem can be solved by Greedy approach
- Travelling salesman problem is an approach to visit n cities exactly once and come back to initial city with minimum expenditure which can be solved by Dynamic programming.
- Maximum element in an array can be found using brute force technique.

---

Q28 For what value of c and $n_0$ $f(n)=O(g(n))$
Consider $f(n)=100n+6$, $g(n)=^{n2}$

A. 106,2
B. 80 ,2
C. 106,1
D. 80,1

**Answer C**

Explanation
$f(n) <= cg(n)$
$100n+6 <= c(n^2)$
$100n/n^2 + 6/n^2 <= c$
$100/n +6/n^2 <= c$
**Let n=0**
$0/0+6/0 <=c$
$0+\infty <=0$ (F)
**Let n=1**
$100 +6 <=c$
$106<=c$
**Assume c=106**
$100n+6 <= 106(n^2)$
**Let n=0**
$0+6 <=0$
$6<=0$ (F)
**Let n=1**
$100 +6 <=106$
$106<=106$ (T)
**Now, f(n)=O(g(n))**
for c=106 and $n_0=1$ $(n>=n_0>=0)$

---

Q29 What is the time complexity of Dijkstra's algorithm?
a) O(N)
b) $O(N^3)$
c) $O(N^2)$
d) O(logN)

**Answer C**

Explanation: Time complexity of Dijkstra's algorithm is $O(N^2)$ because of the use of doubly nested for loops. It depends on how the table is manipulated.

---

Q30 Assume that a merge sort algorithm in the worst case takes 30 seconds for an input of size 64. Which of the following most closely approximates the maximum input size of a problem that can be solved in 6 minutes?

A. 256
B. 512
C. 64
D. 128

**Answer B**

Explanation: Time complexity of merge sort is O(nlogn). So c*64log64 = 30 so c=5/64
For six minutes 5/64*nlogn=6*60. On solving n=512

---

Q31. Which of the problems cannot be solved by backtracking method?

A. N Queen problem
B. Traveling salesman problem
C. Subset sum problem
D. Hamiltonian circuit problem

**Answer B**
Explanation
Travelling salesman problem can be solved by dynamic programming

---

Q32You have to sort a list L, consisting of a sorted list followed by a few 'random' elements. Which of the following sorting method would be most suitable for such a task?

A. Bubble sort
B. Selection sort
C. Quick sort
D. Insertion sort

**Answer D**
**Explanation**

- Bubble sort will take $O(n^2)$ time in best case.
- Selection sort will take $O(n^2)$ time in best case.
- Quick sort will take $O(n^2)$ time in this case because it is the worst case for quick sort.
- Insertion sort will take O(n).

---

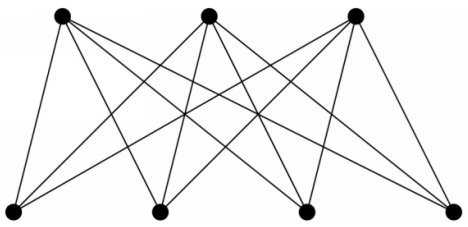
Q33 Which of the following is true?

A. Prim's algorithm can also be used for disconnected graphs
B. Kruskal's algorithm can also run on the disconnected graphs
C. Prims algorithm is simpler than Kruskal's algorithm
D. In Kruskal's sort edges are added to MST in decreasing order of their weight

**Answer B**
**Explanation**
Prim's algorithm iterates from one node to another, so it cannot be applied for disconnected graph. Kruskal's algorithm can be applied to the disconnected graphs to construct the minimum cost forest. Kruskal's algorithm is comparatively easier and simpler than prim's algorithm.

---

Q 34 What will be the chromatic number for a bipartite graph having n vertices?



A bipartite graph

A. 2n
B. 1
C. 2
D. N

**Answer C**

Explanation: A bipartite graph is graph such that no two vertices of the same set are adjacent to each other. So the chromatic number for such a graph will be 2.

---

Q 35 The government wants to connect 10 villages through roads in order to develop that area. Which algorithm do you suggest to the plan manager for efficient implementation?

A. Prism's algorithm
B. Dijkstra's algorithm
C. Travelling salesman
D. Backtracking algorithm

Answer A
Prims algorithm can be used to connect these villages by minimum and optimized way

---

Q36 Consider the following statements.

S1. Kruskal's algorithm might produce a non-minimal spanning tree.
S2. Kruskal's algorithm can efficiently implemented using the disjoint-set data structure.
a) S1 is true but S2 is false
b) Both S1 and S2 are false
c) Both S1 and S2 are true
d) S2 is true but S1 is false
**Answer D**

Explanation: In Kruskal's algorithm, the disjoint-set data structure efficiently identifies the components containing a vertex and adds the new edges. And Kruskal's algorithm always finds the MST for the connected graph.

---

Q 37. In a modified merge sort, the input array is split at a position one-third of the length(N) of the array. Which of the following is the tightest upper bound on time complexity of this modified Merge Sort.

A. N(logN base 3)
B. N(logN base 2/3)
C. N(logN base 1/3)
D. N(logN base 3/2)

**Answer D**

Explanation

The time complexity is given by T(N)=T(N/3) +T(2N/3) +N. On solving the above recurrence relation, we will get N (logN base 3/2)

---

Q 38 Which sorting algorithms is most efficient to sort string consisting of ASCII characters?

A. Quick sort
B. Heap sort
C. Merge sort
D. Counting sort

**Answer D**

Counting algorithm is efficient when range of data to be stored is fixed. In the question the range is from 0 to 255 (ASCII range). Counting sort uses as extra constant space proportional to range of data.

---

Q39. Which of the following is false about Prim's algorithm?

| |
|---|
| A. It is a greedy algorithm |
| B. It constructs MST by selecting edges in increasing order of their weights |
| C. It never accepts cycles in the MST |
| D. It can be implemented using the Fibonacci heap |

**Answer B**

**Explanation**

Prim's algorithm can be implemented using Fibonacci heap and it never accepts cycles. And Prim's algorithm follows greedy approach. Prim's algorithms span from one vertex to another.

---

Q40. What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search?

A. N
B. NlogN
C. N^2
D. N(logN)^2

**Answer C**
Applying binary search to calculate the position of the data to be inserted doesn't reduce the time complexity of insertion sort.
So it will still be N^2