1)MIN MAX USING DIVIDE CONQUER

```
Enter the total number of numbers : 5
Enter the numbers :
23
45
67
98
34
Minimum element in the array : 23
Maximum element in the array : 98
Average of the elements in the array: 53.40
Range of the elements in the array: 75
```

2) STRASSEN MATRIX MULTILPICATION

```
Enter the elements of Matrix A:
12 23 34 45
Enter the elements of Matrix B:
56 67 78 89
Entered Matrix A is:
12      23
34      45
Entered Matrix B is:
56      67
78      89
Resultant Matrix is:
2466    2851
5414    6283
Optimized Resultant Matrix is:
2466    2851
5414    6283
```

3)QuickSort

```
Enter the number of elements: 5
Enter 5 elements:
34
78
12
23
98
Original array: 34 78 12 23 98
Sorted array: 12 23 34 78 98 _
```

4)Merge Sort

```
Enter the number of elements (maximum 11):
5
Enter 5 elements:
12
34
2
78
91
List before sorting:
12 34 2 78 91
List after sorting:
2 12 34 78 91
```

5) Dijkshtra

```
Enter the number of vertices: 5
Enter the adjacency matrix (0 for no connection):
0 2 6 12 15
0 0 7 0 3
0 0 0 5 0
0 0 0 0 3
0 0 0 0 0
Enter the source vertex: 0
Vertex   Distance from Source
0                0
1                2
2                6
3                11
4                5
```

6)Activity Selection

```
Enter the number of task:8
Enter the start time of task : 1
0
1
4
2
5
3
4
Enter the finish time of task : 3
4
2
6
9
8
5
5

Tasks that are selected : 3     7       6
```

7)Fractional knapsack

```
Enter the no. of objects:- 7

Enter the wts and profits of each object 1: 8 4

Enter the wts and profits of each object 2: 6 6

Enter the wts and profits of each object 3: 3 7

Enter the wts and profits of each object 4: 9 8
```

```
Enter the wts and profits of each object 5: 2 1

Enter the wts and profits of each object 6: 4 3

Enter the wts and profits of each object 7: 5 2

Enter the capacity of knapsack:- 15

The result vector is:- 1.000000 1.000000        0.666667        0.000000        0.000000        0.000000        0.000000
Maximum profit is:- 18.333334
```

8)Prisms

```
Enter the adjacency matrix:
0 2 0 6 0
2 0 3 8 5
0 3 0 0 7
6 8 0 0 9
0 5 7 9 0

 Edge    Weight
 0 <-> 1 2
 1 <-> 2 3
 0 <-> 3 6
 1 <-> 4 5
Total Cost = 16_
```

9)Job Sceduling

```
Enter the number of jobs: 4
Enter job ID, profit, and deadline for job 1: 1 100 2
Enter job ID, profit, and deadline for job 2: 2 10 1
Enter job ID, profit, and deadline for job 3: 3 15 2
Enter job ID, profit, and deadline for job 4: 4 27 1
Optimal schedule J: 4 1
Profit set after job scheduling: 27 100
Total profit: 127
```

10)Matrix Chain Multiplication

```
Enter the number of matrices: 4
Enter the dimensions of matrices (including dimensions of result matrix):
p[0]: 5
p[1]: 10
p[2]: 15
p[3]: 20
p[4]: 25
The minimum cost matrix is:
0 750 2250 4750
0 0 3000 8000
0 0 0 7500
0 0 0 0

The split points (k-values) matrix is:
0 1 2 3
0 0 2 3
0 0 0 3
0 0 0 0

Optimal parenthesization is: (((A1A2)A3)A4)
```

11)All PaIr Shortest Path

```
Enter the number of vertices: 4
Enter the edges:
Enter weight for edge [0][1], or 32767 for no edge: 32767
Enter weight for edge [0][2], or 32767 for no edge: -2
Enter weight for edge [0][3], or 32767 for no edge: 32767
Enter weight for edge [1][0], or 32767 for no edge: 4
Enter weight for edge [1][2], or 32767 for no edge: 3
Enter weight for edge [1][3], or 32767 for no edge: 32767
Enter weight for edge [2][0], or 32767 for no edge: 32767
Enter weight for edge [2][1], or 32767 for no edge: 32767
Enter weight for edge [2][3], or 32767 for no edge: 2
Enter weight for edge [3][0], or 32767 for no edge: 32767
Enter weight for edge [3][1], or 32767 for no edge: -1
Enter weight for edge [3][2], or 32767 for no edge: 32767
The original graph is:
   0  INF   -2  INF
   4    0    3  INF
 INF  INF    0    2
 INF   -1  INF    0
The shortest path matrix is:
   0   -1   -2    0
   4    0    2    4
   5    1    0    2
   3   -1    1    0
```

12)0/1 knapsack problem

```
Enter the number of items: 3
Enter the capacities of the items:
Weight 1: 4
Profit 1: 10
Weight 2: 6
Profit 2: 12
Weight 3: 8
Profit 3: 15
Enter the capacity of the knapsack: 10
DP Table:
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 10 10 10 10 10 10 10
0 0 0 0 10 10 12 12 12 12 22
0 0 0 0 10 10 12 12 15 15 22
The maximum profit is 22
```

13)Coin change proble using dynamic

```
Enter the number of coins: 4
Enter the coins: 1 3 5 9
Enter the total amount: 10
Capacity/Coins
          0   1   2   3   4   5   6   7   8   9  10
  1       0   1   2   3   4   5   6   7   8   9  10
  3       0   1   2   1   2   3   2   3   4   3   4
  5       0   1   2   1   2   1   2   3   2   3   2
  9       0   1   2   1   2   1   2   3   2   1   2
Coins used: 5 5
Minimum number of coins required: 2
_
```

14)Coin change problem using Greedy

```
Enter the number of coins: 4
Enter the coins: 1 3 5 9
Enter the total amount: 10
Coins used: 9 1
Minimum number of coins required using greedy approach: 2
```

15)LCS

```
Enter first string: EXAMPLE
Enter second string: APE
LCS of "EXAMPLE" and "APE" is "APE"
Length of LCS: 3
```

16) Bell man ford

```
Enter number of vertices and edges: 5 9
Enter source, destination, and weight for each edge:
0 1 4
0 2 2
1 2 3
2 1 1
1 3 2
4 3 -5
2 4 5
1 4 3
2 3 4
Enter source vertex: 0
Vertex   Distance from Source
0                0
1                3
2                2
3                1
4                6
```

17)OBST

```
Enter the number of keys: 4
Enter the keys: 10 20 30 40
Enter the frequencies: 2 4 6 3
Cost of Optimal BST is 26
```

18)Hamiltonian Cycle

```
Enter the number of vertices: 5
Enter the adjacency matrix:
0 1 1 0 1
1 0 1 1 1
1 1 0 1 0
0 1 1 0 1
1 1 0 1 0
Hamiltonian Cycle:
0 1 2 3 4 0
Hamiltonian Cycle:
0 1 4 3 2 0
Hamiltonian Cycle:
0 2 1 3 4 0
Hamiltonian Cycle:
0 2 3 1 4 0
Hamiltonian Cycle:
0 2 3 4 1 0
Hamiltonian Cycle:
0 4 1 3 2 0
Hamiltonian Cycle:
0 4 3 1 2 0
Hamiltonian Cycle:
0 4 3 2 1 0
Total Hamiltonian Cycles found: 8
```

`19)Graph Coloring

```
Enter the number of colors: 3
Enter the adjacency matrix (4 x 4):
0 1 0 1
1 0 1 0
0 1 0 1
1 0 1 0
Vertex Colors: 1 2 1 2
Vertex Colors: 1 2 1 3
Vertex Colors: 1 2 3 2
Vertex Colors: 1 3 1 2
Vertex Colors: 1 3 1 3
Vertex Colors: 1 3 2 3
Vertex Colors: 2 1 2 1
Vertex Colors: 2 1 2 3
Vertex Colors: 2 1 3 1
Vertex Colors: 2 3 1 3
Vertex Colors: 2 3 2 1
Vertex Colors: 2 3 2 3
Vertex Colors: 3 1 2 1
Vertex Colors: 3 1 3 1
Vertex Colors: 3 1 3 2
Vertex Colors: 3 2 1 2
Vertex Colors: 3 2 3 1
Vertex Colors: 3 2 3 2
```

20)N queen

```
Enter the number of queens (Max 20): 4
0 0 1 0
1 0 0 0
0 0 0 1
0 1 0 0

0 1 0 0
0 0 0 1
1 0 0 0
0 0 1 0

Total number of solutions: 2
```

21)Rabin Karb

```
Enter the text: CCACCAACDAB
Enter the pattern to search: DAB
Pattern found at index 8
```

Or take heyhihey

22)Naïve string Matching

```
Enter the text: AABAACAADAABAAABAA
Enter the pattern: AABA
Text: AABAACAADAABAAABAA
Pattern: AABA
Pattern found at following indices:
Pattern found at index 0
Pattern found at index 9
Pattern found at index 13
```

23)KMP

```
Enter the text: AABAACAADAABAAABAA
Enter the pattern: AABA
Pattern found at index 0
Pattern found at index 9
Pattern found at index 13
Pattern not found in text/Pattern Ended
```