Design & Analysis Of Algorithm

Lab Experiment -9

NAME : SURIYAPRAKASH.C

ROLL NO:CH.EN.U4CSE20170

SUBJECT: DESIGN & ANALYSIS OF ALGORITHM

SUBJECT CODE: 19CSE302

Submitted to – Mrs. Ashwini,

Department of CSE,

ASE Chennai campus.

EX NO:9

Knapsack algorithm.

AIM:

To write an algorithm to implement Knapsack algorithm.

ALGORITHM:

1) Input no. of objects and sack capacity in n & sack Cap respectively.
2) Input Profit and Weight arrays respectively.
3) Define base condition
4) Check if nth item's weight is more than sack capacity, then exclude nth item from the optimal solution.
5) else print the maximum of the 2 cases: Nth item included, Nth item not included.

CODE SCREEN:

```
def knapsack(arr,m):
    n=len(arr)
    size=m
    for i in range(n):
        ppw=arr[i][1]/arr[i][2]
        ppw=round(ppw,2)
        arr[i].append(ppw)
    for i in range(n):
        for j in range(n-i-1):
            if arr[j][3]<arr[j+1][3]:
                arr[j],arr[j+1]=arr[j+1],arr[j]
    x={}
    profit=0
    for i in range(n):
        if size>=arr[i][2]:
            size=size-arr[i][2]
            x[arr[i][0]]=1
            profit+=arr[i][2]*arr[i][3]
        elif size!=0:
            val=size/arr[i][2]
            x[arr[i][0]]=val
```

```
            profit+=size*arr[i][3]
            size=0
        else:
            x[arr[i][0]]=0
    print(profit)
    print(x)
    print(arr)
if __name__ == '__main__':
    arr = [['o1', 10, 2],
              ['o2', 5, 3],
              ['o3', 15, 5],
              ['o4', 7, 7],
              ['o5', 6, 1],
              ['o6', 18, 4],
              ['o7', 3, 1]]
    print("Suriyaprakash- 20170 \n-------------")
    print("maximum profit sequence is :")
    knapsack(arr, 15)
```

OUTPUT SCREEN :

```
PS D:\python> & C:/Users/HP/AppData/Local/Programs/Python/Python310/python.exe d:/python
/DAA/knapsack.py
Suriyaprakash- 20170
-------------
maximum profit sequence is :
55.34
{'o5': 1, 'o1': 1, 'o6': 1, 'o3': 1, 'o7': 1, 'o2': 0.6666666666666666, 'o4': 0}
[['o5', 6, 1, 6.0], ['o1', 10, 2, 5.0], ['o6', 18, 4, 4.5], ['o3', 15, 5, 3.0], ['o7', 3
, 1, 3.0], ['o2', 5, 3, 1.67], ['o4', 7, 7, 1.0]]
PS D:\python>
```

TIME COMPLEXITY:

O (n W)

Worst case -O(N*W)

RESULT:

I have studied and understood the Knapsack in python language and executed the program successfully.

THANK YOU !!