

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 11

**Institute of Computer Technology**  
**B. Tech Computer Science and Engineering**

**Sub: Algorithm Analysis and Design**

**Practical 11**

**Problem** : A thief carrying a single knapsack with limited ( $W = 5$ ) capacity. The museum you stole had ( $n=4$ ) artifacts that you could steal. Unfortunately, you might not be able to steal the entire artifact because of your limited knapsack capacity. Help the thief to cherry pick the artifact in order to maximize the total value ( $\leq W$ ) of the artifacts you stole.

First solve the given below example:

Let  $n = 4$ ,  $W=5$

$(P1, P2, P3, P4) = (3,4,5,6)$

$(w1, w2, w3, w4) = (2,3,4,5)$

**Code :**

```
import YSL_io

def knapsack(W, wt, val, n):
    k = [[0 for x in range(W + 1)] for x in range(n + 1)]

    for i in range(n + 1):
        for w in range(W + 1):
            if i == 0 or w == 0:
                k[i][w] = 0
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 11

```
elif wt[i - 1] ≤ w:
    k[i][w] = max(val[i - 1] + k[i - 1][w - wt[i - 1]], k[i - 1][w])
else:
    k[i][w] = k[i - 1][w]

YSL_io.printBLU("\nMaximum profit : ", end='')
print(k[n][W])

YSL_io.printRED("\nKnapsack Matrix : ")
for i in range(n + 1):
    for j in range(W + 1):
        YSL_io.printRED(f'{k[i][j]} ', end='')
    print()

selected = []
res = k[n][W]
w = W
i = n
while i > 0 and res > 0:
    if res == k[i - 1][w]:
        i -= 1
    else:
        selected.append(wt[i - 1])
        res -= val[i - 1]
        w -= wt[i - 1]
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 11

```
i -= 1

YSL_io.printORNG("\nSelected knapsack(s) : ", end='')
print(selected)


n = int(YSL_io.inputGRN("Number of entries (n) : "))
print()

W = int(YSL_io.inputGRN('Maximum capacity (W) : '))
print()

val = []
wt = []

for i in range(n):
    val.append(int(YSL_io.inputMGNTA(f"Enter the price {i+1} : ")))

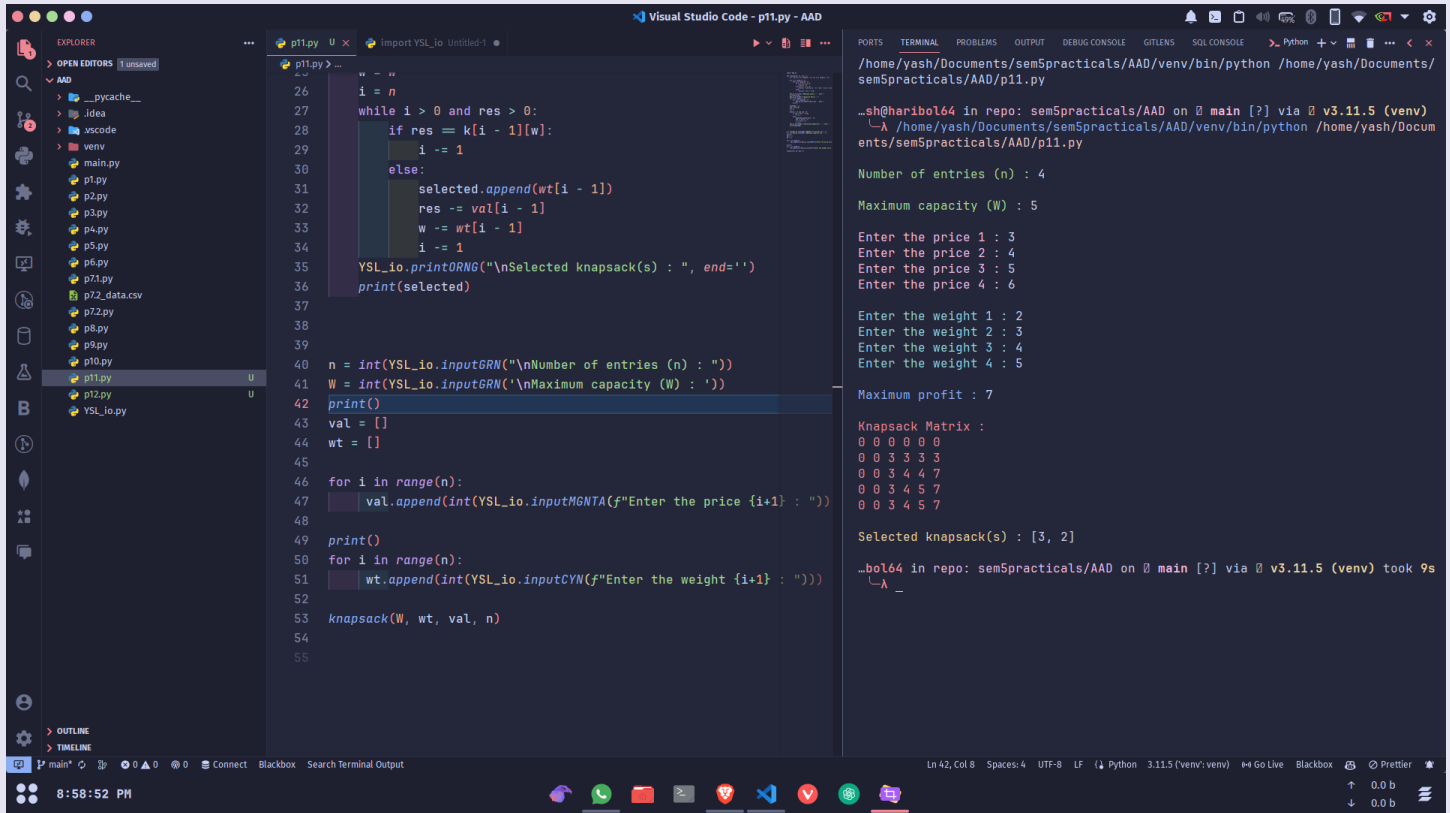
print()

for i in range(n):
    wt.append(int(YSL_io.inputCYN(f"Enter the weight {i+1} : ")))

knapsack(W, wt, val, n)
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA Batch - 51  
AAD Practical 11

## Screenshot:



The screenshot displays a Visual Studio Code editor window titled "Visual Studio Code - p11.py - AAD". The editor shows a Python script named "p11.py" with the following code:

```
26 w = []
27 i = n
28 while i > 0 and res > 0:
29     if res == k[i - 1][w]:
30         i -= 1
31     else:
32         selected.append(wt[i - 1])
33         res -= val[i - 1]
34         w += wt[i - 1]
35         i -= 1
36 YSL_io.printDRNG("\nSelected knapsack(s) : ", end='')
37 print(selected)
38
39 n = int(YSL_io.inputGRN("\nNumber of entries (n) : "))
40 W = int(YSL_io.inputGRN("\nMaximum capacity (W) : "))
41 print()
42 val = []
43 wt = []
44
45 for i in range(n):
46     val.append(int(YSL_io.inputMGNTA(f"Enter the price {i+1} : ")))
47
48 print()
49 for i in range(n):
50     wt.append(int(YSL_io.inputCYN(f"Enter the weight {i+1} : ")))
51
52 knapsack(W, wt, val, n)
```

The terminal output on the right shows the execution of the script:

```
/home/yash/Documents/sem5practicals/AAD/venv/bin/python /home/yash/Documents/sem5practicals/AAD/p11.py
...sh@haribol64 in repo: sem5practicals/AAD on  main (?) via  v3.11.5 (venv)
└─ /home/yash/Documents/sem5practicals/AAD/venv/bin/python /home/yash/Documents/sem5practicals/AAD/p11.py

Number of entries (n) : 4
Maximum capacity (W) : 5

Enter the price 1 : 3
Enter the price 2 : 4
Enter the price 3 : 5
Enter the price 4 : 6

Enter the weight 1 : 2
Enter the weight 2 : 3
Enter the weight 3 : 4
Enter the weight 4 : 5

Maximum profit : 7

Knapsack Matrix :
0 0 0 0 0 0
0 0 3 3 3 3
0 0 3 4 4 7
0 0 3 4 5 7
0 0 3 4 5 7

Selected knapsack(s) : [3, 2]

...bol64 in repo: sem5practicals/AAD on  main (?) via  v3.11.5 (venv) took 9s
└─ _
```