

Lab eval

```
# SET 3
# You are on a treasure hunt within a mysterious castle, which
# has a huge collection of valuables. You are getting an
# opportunity to enter the castle to take the valuables
# worth '$'. Each valuable has varying values, and you are
# trying to achieve a total value exactly equal to '$'. In
# the exit of the castle, you will be checked by a guard, if
# you have collected the minimum number of valuables from
# the castle and ensure their total value is exactly '$'.
# Then it is declared that you have won the game.

# My solution : This problem is very similar to the knapsack problem. Just how we maximize the value here we minimize
# the number of items chosen
def get_valuables(amt, val):
    L = [[0] * (amt + 1) for j in range(len(val) + 1)]
    for i in range(len(val) + 1):
        for j in range(amt + 1):
            if i == 0 or j == 0:
                L[i][j] = 0

            elif val[i-1] <= j:
                L[i][j] = 1 + L[i-1][j - val[i-1]]

            else:
                L[i][j] = L[i-1][j]
    print(L)
    return L[len(val)][amt]

if __name__ == "__main__":
    print("This is the solution for the question in set 3")
    V = int(input("Enter the amount to be collected from the castle :"))
    n = int(input("Enter the number of valuables"))
    value = []
    for i in range(n):
        a = int(input())
        value.append(a)
    print(get_valuables(V, value))

#testcases tried
#value = [1, 2, 3, 4, 5, 6] V = 10 :- answer:3
# value = [1, 2, 3, 4] V = 6 :- answer:2
# value = [1, 2, 3] V = 3 :- answer:1
```

```
/Users/surya/anaconda3/envs/tf/bin/python /Users/surya/Documents/Surya/Git/Repo1/Data structures/Lab Eval 2/21461_eval.py
This is the solution for the question in set 3
Enter the amount to be collected from the castle :6
Enter the number of valuables:3
1
2
3
[[0, 0, 0, 0, 0, 0, 0], [0, 1, 1, 1, 1, 1, 1], [0, 1, 1, 2, 2, 2, 2], [0, 1, 1, 1, 2, 2, 3]]
3
```

```
/Users/surya/anaconda3/envs/tf/bin/python /Users/surya/Documents/Surya/Git/Repo1/Data structures/Lab Eval 2/21461_eval.py
This is the solution for the question in set 3
Enter the amount to be collected from the castle :10
Enter the number of valuables6
1
2
3
4
5
6
[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 1, 1, 1, 1, 1, 1, 1, 1, 1], [0, 1, 1, 2, 2, 2, 2, 2, 2, 2], [0, 1, 1, 1, 2, 2, 3, 3, 3, 3], [0, 1, 1, 1, 1, 2, 2, 2, 3, 2]

Process finished with exit code 0
|
```

```
f __name__ == "__main__":
    for i in range(n):
        # ... (code for generating the array) ...

Run 21461_eval x
:
/Users/surya/anaconda3/envs/tf/bin/python /Users/surya/Documents/Surya/Git/Repo1/Data structures/Lab Eval 2/21461_eval.py
This is the solution for the question in set 3
Enter the amount to be collected from the castle :5
Enter the number of valuables3
1
2
3
[[0, 0, 0, 0, 0, 0], [0, 1, 1, 1, 1, 1], [0, 1, 1, 2, 2, 2], [0, 1, 1, 1, 2, 2]]
2

Process finished with exit code 0
|
```