Name : Shashank K

ID : SE21UCSE198

Lab Submission 5

I have chosen the fractional knapsack problem and implemented the function **"knapsack"** in python.

Overview:

I have given three inputs as dictionaries. The structure of dictionary follows {value:weight}. The function accepts a dictionary and a weight value as parameters.

The function first sorts the dictionary based on the ratio value:weight in descending order. It then starts choosing the values with highest ratios.

The complexity of **sorted** in python is O(nlogn). The complexity of the traversal of dictionary is n at max so the max complexity would be O(nlogn) if we consider the sorting complexity too. The space complexity is O(n).

Inputs :

```
example1 = {50:10,60:20,80:30}
W1 = 40
example2 = {10: 2,5: 3,15: 5,7: 7,6: 1}
W2 = 10
example3 = {100: 20,60: 10,120: 40,50: 30}
W3 = 50
```

Output:

```
shash@LAPTOP-74LO6U0H MINGW64 /c/Mahindra Notes and schedule/semester 5/DAA/Assignment week 5 (main)
$ python fractional_knapsack.py
Profit for {50: 10, 60: 20, 80: 30} is 136.66666666666666
Profit for {10: 2, 5: 3, 15: 5, 7: 7, 6: 1} is 34.333333333333336
Profit for {100: 20, 60: 10, 120: 40, 50: 30} is 220.0
```

Cross check on paper