INSTITUTE OF COMPUTER TECHNOLOGY

B-TECH COMPUTER SCIENCE ENGINEERING 2025-26

SUBJECT:-Algorithm Analysis and Design

NAME: Rahul Prajapati

ENRLL. NO: 23162171020

BRANCH: CYBER SECURITY

BATCH: 52

PRACTICAL_8

Aim:

- A thief is robbing a store and can carry a maximal weight of W into his knapsack. There are n items available in the store and weight of i^{th} item is wi and its profit is pi. What items should the thief take?
- In this context, the items should be selected in such a way that the thief will carry those items for which he will gain maximum profit. Hence, the objective of the thief is to maximize the profit.
- Implement Program for fractional knapsack using Greedy design technique.
  Note: First solve the example:
  W=60

Item	A	B	C	D
Profit	280	100	120	120
Weight	40	10	20	24

Sample Input:-
p=[280,100,120,120]
w=[40,10,20,24]
W=60

Sample Output:-
Profit [100, 280, 120, 120]
Weight [10, 40, 20, 24]
Ratio [10.0, 7.0, 6.0, 5.0]
[1, 1, 0.5, 0]
Total profit : 440.0

CODE:

```python
def fractional_knapsack(p, w, W):
    items = [(p[i], w[i], p[i]/w[i]) for i in range(len(p))]

    items.sort(key=lambda x: x[2], reverse=True)

    total_profit = 0
    fractions = []

    for profit, weight, ratio in items:
        if W == 0:
            break

        if weight <= W:
            total_profit += profit
            fractions.append(1)
            W -= weight
        else:
            frac = W / weight
            total_profit += profit * frac
            fractions.append(frac)
            W = 0

    print("Fractions taken:", fractions)
    print("Total Profit:", total_profit)

# Example
p = [280, 100, 120, 120]
w = [40, 10, 20, 24]
W = 60
fractional_knapsack(p, w, W)
```

OUTPUT:

```
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES> python .\Practical8_1.py
Fractions taken: [1, 1, 0.5]
Total Profit: 440.0
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES>
```