

Practical No. 3

Aim : Write a program to solve a fractional Knapsack problem using a greedy method.

Code :

```
def fractional_knapsack(values, weights, capacity):
    ratio = [(v / w, w, v, i) for i, (v, w) in enumerate(zip(values, weights))]
    ratio.sort(key=lambda x: x[0], reverse=True)

    total_value = 0.0
    fractions = [0] * len(values)

    for r, w, v, i in ratio:
        if capacity == 0:
            break
        amount = min(w, capacity)
        fractions[i] = amount / w
        total_value += amount * r
        capacity -= amount

    return total_value, fractions

def main():
    n = int(input("Enter the number of items: "))
    values = []
    weights = []
    for i in range(n):
        v = float(input(f"Enter value of item {i + 1}: "))
        w = float(input(f"Enter weight of item {i + 1}: "))
        values.append(v)
        weights.append(w)
    capacity = float(input("Enter the capacity of knapsack: "))

    max_value, fractions = fractional_knapsack(values, weights, capacity)

    print(f"\nMaximum value achievable: {max_value:.2f}")
    print("Fractions of items taken:")
    for i, frac in enumerate(fractions):
        print(f"Item {i + 1}: {int(frac * 100)}%")

if __name__ == "__main__":
    main()
```

Output :

Enter the number of items: 6
Enter value of item 1: 60
Enter weight of item 1: 10
Enter value of item 2: 100
Enter weight of item 2: 20
Enter value of item 3: 120
Enter weight of item 3: 30
Enter value of item 4: 80
Enter weight of item 4: 40
Enter value of item 5: 30
Enter weight of item 5: 5
Enter value of item 6: 50
Enter weight of item 6: 25
Enter the capacity of knapsack: 60

Maximum value achievable: 290.00

Fractions of items taken:

Item 1: 100%
Item 2: 100%
Item 3: 83%
Item 4: 0%
Item 5: 100%
Item 6: 0%