LABORATORY REPORT

Algorithm Laboratory (CS-39001)

B.Tech Program in ECSc

Submitted By

Name:- SANNIDHI DEB

Roll No: 2330044

Kalinga Institute of Industrial Technology

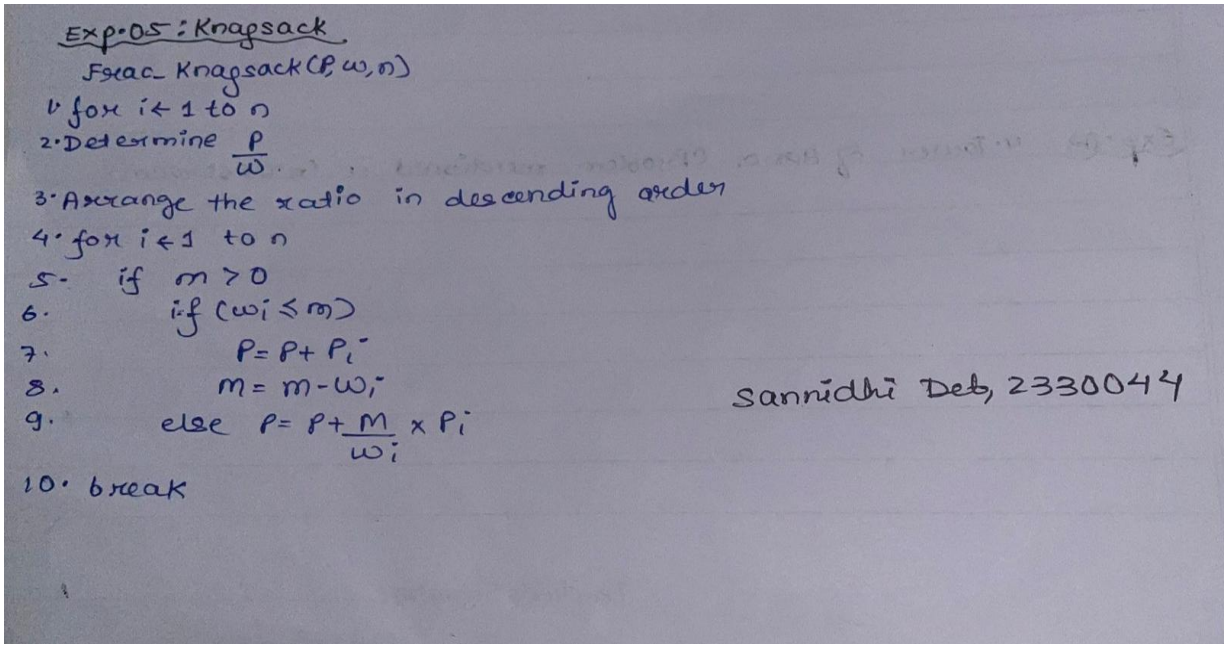
(Deemed to be University) Bhubaneswar, India

Autumn, 2025

Table of Contents

Experiment Number	5.1
Experiment Title	Write an algorithm and a C program to find the maximum profit nearest to but not exceeding the given knapsack capacity using the Fractional Knapsack algorithm. Input: Number of items: 3 Weights of items, {w1, w2, w3} = {18, 15, 10} Profits of items, {P1, P2, P3} = {25, 24, 15} Maximum capacity of knapsack, m =20
Date of Experiment	18/09/2025
Date of Submission	24/09/2025

1.Algorithm:-



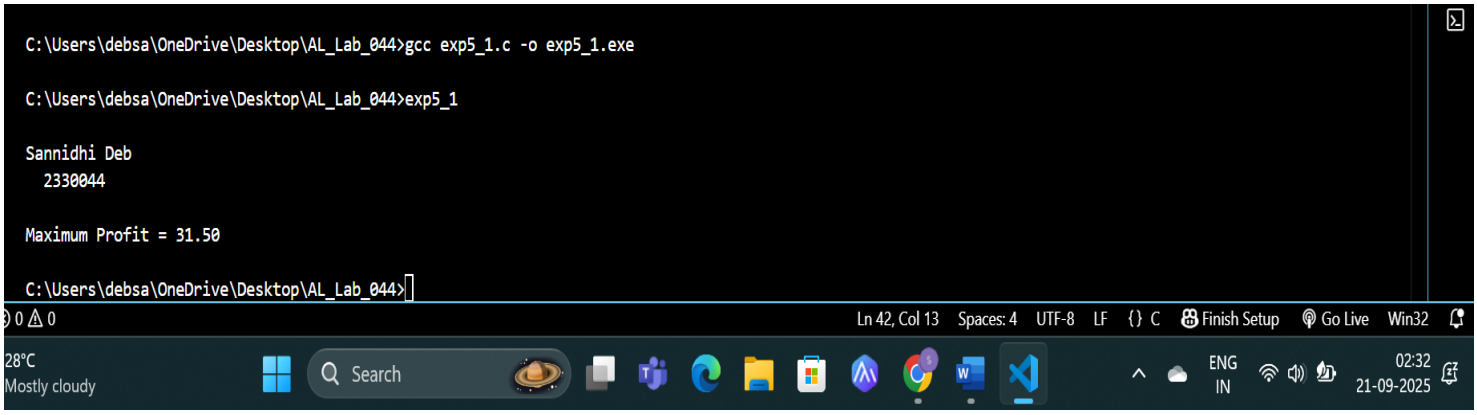
2. Code:-

```
#include <stdio.h>

typedef struct {
    int weight;
    int profit;
    float ratio;
} Item;

void sortItems(Item items[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            if (items[i].ratio < items[j].ratio) {
                Item temp = items[i];
                items[i] = items[j];
                items[j] = temp;
            }
        }
    }
}

int main() {
    int n = 3;
    int m = 20;
    int weights[] = {18, 15, 10};
```

```
int profits[] = {25, 24, 15};

Item items[n];

for (int i = 0; i < n; i++) {
    items[i].weight = weights[i];
    items[i].profit = profits[i];
    items[i].ratio = (float)profits[i] / weights[i];
}

sortItems(items, n);

int remaining_capacity = m;
float total_profit = 0.0;

for (int i = 0; i < n; i++) {
    if (items[i].weight <= remaining_capacity) {

        total_profit += items[i].profit;
        remaining_capacity -= items[i].weight;
    } else {

        total_profit += items[i].profit * ((float)remaining_capacity / items[i].weight);
        remaining_capacity = 0;
        break;
```

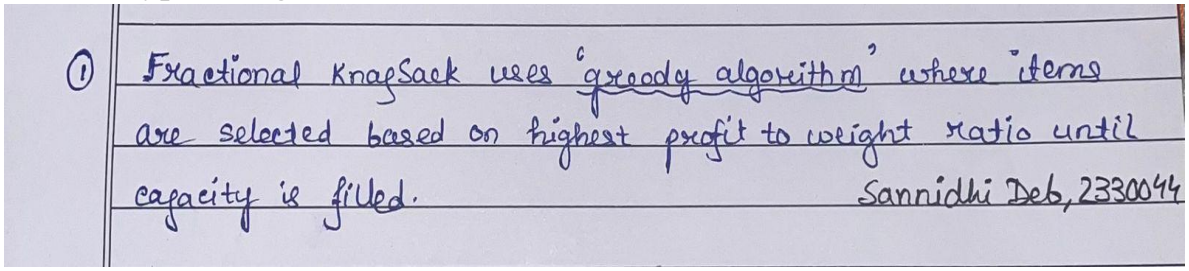
```
    }
  }
  printf("\nSannidhi Deb\n  2330044\n\n");
  printf("Maximum Profit = %.2f\n", total_profit);

  return 0;
}
```

3.Results/Output:- Entire Screen Shot including Date & Time:-

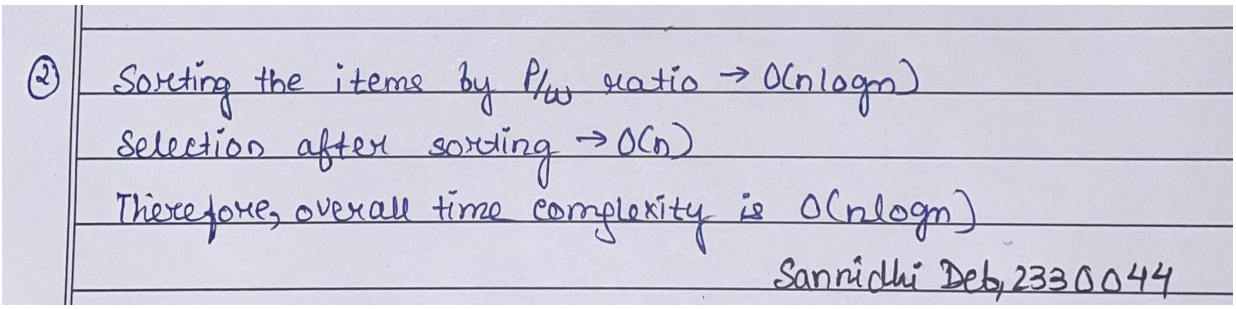


4. Remarks:-

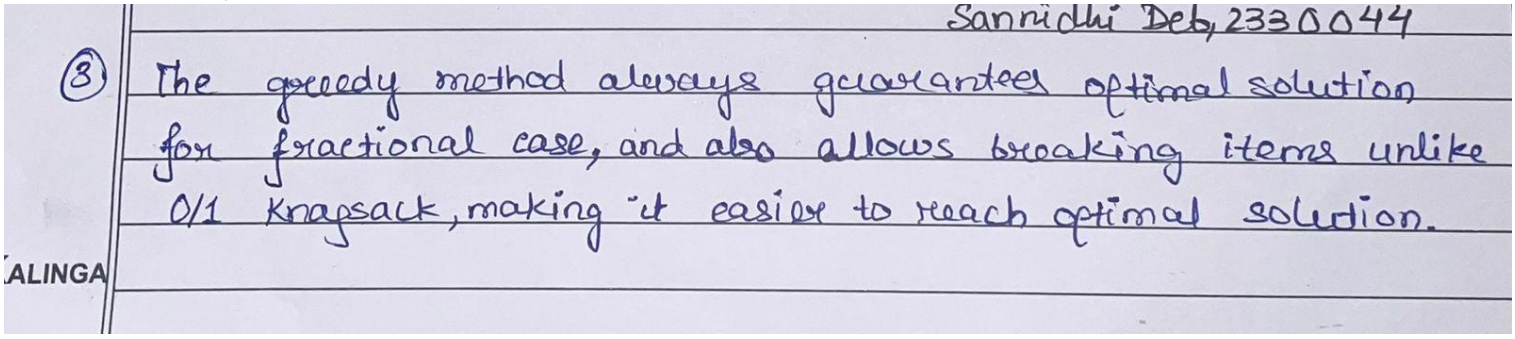
1. What type of algorithm is used?



2. Analyze the complexity of your algorithm.

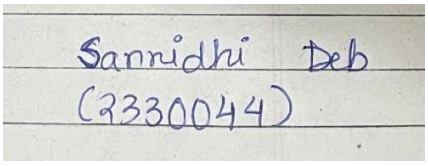


3.Any other observations?



5. Conclusion:-

The fractional knapsack experiment successfully demonstrated how greedy optimization can be applied to maximize profit without exceeding the capacity constraint. By prioritizing items based on profit-to-weight ratio and allowing fractional selection, the algorithm achieved an optimal solution efficiently, highlighting its effectiveness in resource allocation problems



Signature of the FIC

Sannidhi Deb

(Name of the FIC)