

## Question 01

1. Ramani decides to buy fruits and vegetables using the promotion lottery she won.

She can buy goods up to 30 kg weight in total free of charge.

She can only select an item upto its maximum weight or the price/weight ratio as shown below.

Item No	Item	Weight (Kg)	Price (Rs.)	Price/Weight
01	Apple	1	1050	1050
02	Banana	6	600	100
03	Papaya	8	400	50
04	Watermelon	5	1500	300
05	Orange	4	800	200
06	Potato	6	1800	300
07	Carrot	1	300	300
08	Eggplant	2	900	450
09	Beetroot	3	600	200
10	Ladies' Fingers	5	1000	200

This problem can be solved using the greedy approach in such a way that the maximum price is calculated at the end of selection.

In order for that, we can select the items by,

- Most price
- Most number of items (least weight)
- Most value density (price/weight)

Here, the solution will be created by using the third way which is to select the item with highest price/weight ratio at each step until the weight limit is reached.

2. Here, the main objective of Ramani is to obtain the maximum total price at the end of selection. Since the condition of lottery does not have a limit to price, it is optimal and greedy to select the item with highest price/weight ratio at each step until the weight limit is reached. Here, an item can be selected as fractions if the maximum weight cannot be included. This method represents the fractional knapsack problem. Hence, maximum outcome can be obtained if the price/weight ratio is taken to consideration when choosing items. Therefore, this approach is selected to solve the problem.

3. Constraints –

Weight of the total selection should be limited upto 30 kg

She can only select an item at its maximum weight provided or fractions of it

Item with highest price/weight ratio is selected at each step

4.

```
items = [(1050,1,"Apple"), (600,6,"Banana"), (400,8,"Papaya"), (1500,5,"Watermelon"),  
(800,4,"Orange"), (1800,6,"Potato"), (300,1,"Carrot"), (900,2,"Egg Plant"), (600,3,"Beetroot"),  
(1000,5,"Ladies' Fingers")]
```

```
maxW = 30
```

```
n = len(items)
```

```
def lottery(maxW, items, n)
```

```
    sorted = sort.items() //Sort items in list in descending order of price/weight
```

```
    prices[], weights[] //Create separate lists for prices and weights
```

```
    for i in range(n)
```

```
        if maxW == 0
```

```
            return total
```

```
if weights[i] <= maxW

    maxW = maxW - weights[i]

    total = total + prices[i]

else

    total = total + prices[i] * maxW/weights[i]

    maxW = 0

return total
```