Fractional Knapsack Problem

Given weights and values of n items, we need to put these items in a knapsack of capacity W to get the maximum total value in the knapsack.

In **Fractional Knapsack**, we can break items for maximizing the total value of knapsack. This problem in which we can break an item is also called the fractional knapsack problem.

```
Input:
    Items as (value, weight) pairs
    arr[] = {{60, 10}, {100, 20}, {120, 30}}
    Knapsack Capacity, W = 50;

Output:
    Maximum possible value = 240 (with 0-1 Knapsack, max = 220)
    By taking full items of 10 kg, 20 kg and
    2/3rd of last item of 30 kg
```

A brute-force solution would be to try all possible subset with all different fraction but that will be too much time taking.

An efficient solution is to use Greedy approach. The basic idea of the greedy approach is to calculate the ratio value/weight for each item and sort the item on basis of this ratio. Then take the item with the highest ratio and add them until we can’t add the next item as a whole and at the end add the next item as much as we can. Which will always be the optimal solution to this problem.

Time Complexity: As main time taking step is sorting, the whole problem can be solved in O(n log n) only.