- If there is an intersection between current and last interval then we can just update the last interval's end.
- If current interval is not inside the last interval, and has no intersection with last interval, then we add it into the result list

Fractional knapsack problem :

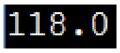
Q3. Given the weights and values of N items, in the form of {value, weight} put these items in a knapsack of capacity W to get the maximum total value in the knapsack. In Fractional Knapsack, we can break items for maximizing the total value of the knapsack.
Input: arr[] = {{60, 10}, {100, 20}, {120, 30}}, W = 50
Output: 240
Explanation: By taking items of weight 10 and 20 kg and 2/3 fraction of 30 kg.
Hence total price will be 60+100+(2/3)(120) = 240

Input: arr[] = {{500, 30}}, W = 10
Output: 166.667

solution :

Code : LP_code3.java

Output :



```
118.0
```

Approach :

- For each item, compute its value / weight ratio.
- Arrange all the items in decreasing order of their value / weight ratio.
- Start putting the items into the knapsack beginning from the item with the highest ratio.
- Put as many items as you can into the knapsack.

Next class teaser:

- Tree