**package** GreedyAlgorithmLab;

**import** java.util.ArrayList;

**import** java.util.Collections;

**import** java.util.HashMap;

**import** java.util.Map;

**class** minWeightInKnapsack **implements** Comparable<minWeightInKnapsack>

{

**public int itemName**;

**public int weight**;

**public int profit**;

**public** minWeightInKnapsack(**int** itemName, **int** weight, **int** profit) {

**this**.**itemName** = itemName;

**this**.**weight** = weight;

**this**.**profit** = profit;

}

@Override

**public int** compareTo(minWeightInKnapsack o) {

**if** (**this**.**weight**>o.**weight**) **return** 1;

**else return** -1;

}

}

**class** TotalProfitMinWeight

{

**public** HashMap<Integer,Double> **hashMap**;

**public** ArrayList<minWeightInKnapsack> **arrayList**;

**public int maxCapacity**;

**public double totalProfit**;

**public** TotalProfitMinWeight(ArrayList<minWeightInKnapsack> arrayList, **int** maxCapacity) {

**this**.**arrayList** = arrayList;

**this**.**maxCapacity** = maxCapacity;

**this**.**totalProfit** = 0;

**this**.**hashMap** = **new** HashMap<>();

}

**public void** giveItems()

{

**for** (minWeightInKnapsack give : **arrayList**)

{

calculateProfit(give.**itemName**, give.**profit**, give.**weight**);

}

}

**public void** calculateProfit(**int** itemName, **int** profit, **int** weight)

{

**if** (**maxCapacity** == 0) **return**;

**else if** (**maxCapacity**!=0)

{

**maxCapacity** = **maxCapacity** - weight;

**if** (**maxCapacity**<0)

{

**maxCapacity** = **maxCapacity** + weight;

**double** fraction = Double.*valueOf*(profit)\*(Double.*valueOf*(**maxCapacity**)/Double.*valueOf*(weight));

**totalProfit** = **totalProfit** + fraction;

**hashMap**.put(itemName,Double.*valueOf*(fraction));

**maxCapacity** = 0;

}

**else if** (**maxCapacity**>0)

{

**totalProfit** = **totalProfit** + profit;

**hashMap**.put(itemName, Double.*valueOf*(profit));

}

}

}

**public void** print()

{

**for** (Map.Entry<Integer, Double> print : **hashMap**.entrySet())

{

**int** key = print.getKey();

**double** profit = print.getValue();

System.***out***.println(**"Item number "**+key+**", and the profit is "**+profit);

}

System.***out***.println(**"---------------------------------------------"**);

System.***out***.println(**"Total profit using min weight = "**+**totalProfit**);

}

}

**public class** knapsackUsingMinimumWeight {

**public static void** main(String[] args) {

**int**[] weight = {1,3,5,7};

**int**[] profit = {5,10,15,9};

ArrayList<minWeightInKnapsack> arrayList = **new** ArrayList<>();

**for** (**int** i=0;i<profit.**length**;i++)

{

arrayList.add(**new** minWeightInKnapsack((i+1),weight[i],profit[i]));

}

Collections.*sort*(arrayList);

**int** maxCapacity = 15;

TotalProfitMinWeight knapsack = **new** TotalProfitMinWeight(arrayList, maxCapacity);

knapsack.giveItems();

knapsack.print();

}

}