

Evan

Only let oneself become strong enough, good enough, can afford the life that you want to.

目录视图

摘要视图

RSS 订阅

程序员，为什么写不好一份简历？ 征文 | 你会为 AI 转型么？ 赠书：7月大咖新书机器学习/Android/python

蓝桥杯 算法提高 快乐司机

标签：acm java 算法

2015-12-10 13:41 1050人阅读 评论(0) 收藏 举报

分类： 蓝桥杯题目 (3)

版权声明：本文为博主原创文章，未经博主允许不得转载。

问题描述

"嘟嘟嘟嘟嘟嘟"

喇叭响

我是汽车小司机

我是小司机

我为祖国运输忙

运输忙"

这是儿歌“快乐的小司机”。话说现在当司机光有红心不行，还要多拉快跑。多拉不是超载，是要让所载货物价值最大，特别是在当前油价日新月异的时候。司机所拉货物为散货，如大米、面粉、沙石、泥土.....

现在知道了汽车核载重量为w，可供选择的物品的数量n。每个物品的重量为gi,价值为pi。求汽车可装载的最大价值。

(n<10000,w<10000,0<gi<=100,0<=pi<=100)

输入格式

输入第一行为由空格分开的两个整数n w

第二行到第n+1行，每行有两个整数，由空格分开，分别表示gi和pi

输出格式

最大价值（保留一位小数）

样例输入

5 36

99 87

68 36

79 43

75 94

7 35

样例输出

71.3

解释：

先装第5号物品，得价值35，占用重量7

再装第4号物品，得价值36.346,占用重量29

最后保留一位小数，得71.3

[java]

```
01. <span style="font-size:14px;">import java.util.Arrays;
02. import java.util.Scanner;
03.
04. public class Main{
05.     static class Values implements Comparable<Values>{
06.         public int num;
07.         public double value;
08.         public Values(int num,double value){
09.             this.num=num;
10.             this.value=value;
11.         }
12.         //实现接口方法
13.         @Override
14.         public int compareTo(Values o) {
15.             if(o instanceof Values){
16.                 double cmp =value - o.value;
17.                 if(cmp>0){
18.                     return 1;
19.                 }else if(cmp<0){
20.                     return -1;
21.                 }else {
22.                     return 0;
23.                 }
24.             }
25.             return 0;
26.         }
27.     }
28.     public static void main(String[] args){
29.         Scanner sc = new Scanner(System.in);
30.         int n = sc.nextInt();
31.         int w = sc.nextInt();
32.         int[] gi = new int[n];
33.         int[] pi = new int[n];
34.         Values[] values = new Values[n];
35.         double sum=0.0;
36.         for(int i=0;i<n;i++){
37.             gi[i]=sc.nextInt();
38.             pi[i]=sc.nextInt();
39.             values[i]=new Values(gi[i], (double)pi[i]/(double)gi[i]);
40.         }
41.         Arrays.sort(values);
42.         for(int i=n-1;i>=0;i--){
43.             if(w>values[i].num){
44.                 sum+=values[i].num*values[i].value;
45.                 w=w-values[i].num;
46.             }else{
47.                 sum+=w*values[i].value;
48.                 break;
49.             }
50.         }
51.         String result = String.format("%.1f", sum);
52.         System.out.println(result);
53.     }
54. }</span>
```

顶 踩
1 0