**2627.直线取最值**

**(lines.cpp/c/pas)**

【问题描述】

在平面直角坐标系里，横坐标为x，纵坐标为y。有n条直线，第i条直线的形式为：y=b[i]+k[i]\*x。其中整数b[i]为截距，整数k[i]为斜率。已知k[i]随着i的增加而减小。另外有m个从小到大排布的横坐标，第i个为x[i]。请求出在每个横坐标位置上各个直线的纵坐标最小值是多少？

【输入格式】输入文件lines.in

输入第一行为正整数n和m，n<=50000，m<=100000。第二行为n个数，代表b[i], 绝对值均不超过100000000。第三行为n个数，代表k[i], 绝对值均不超过100000000。第四行为m个数，代表x[i], 绝对值均不超过100000000。

【输出格式】输出文件lines.out

输出一行，包含m个正整数，有空格隔开。

【输入输出样例1】

|  |  |
| --- | --- |
| lines.in | lines.out |
| 2 3  0 5  0 -1  -2 5 6 | 0 0 -1 |

【说明】注意两条直线交点坐标为(5,0)。

**2628.书籍叠放1**

**(books.cpp/c/pas)**

【问题描述】

你有n本书，编号1到n，需要分几堆叠放在桌面上。每本书的正面都是一个长方形，共四条边，左右两侧边长对应长方形的长度，上下两侧边长对应长方形的宽度。第i本书的长度为h[i]，宽度为w[i]。你可以选择任意几本书分出任意的堆数。但是，叠放的时候要注意，每本书两两之间都保持上下左右四条边平行，不能旋转。每组叠放好的书都会在桌面上占据一定的面积。例如我们将3本书叠放在一堆里：第一本长度为1宽度为4，第二本长度为2宽度为3，第三本长度为3宽度为2，这三本占据的面积为长度为3宽度为4的长方形面积，也就是3\*4=12。

目前，已知你的书籍满足一种特殊顺序：你发现随着编号i增加，书的长度h[i]增加，宽度w[i]减少。请问，经过合理叠放后，总体占据的面积最小是多少？

【输入格式】输入文件books.in

输入第一行为正整数n，n<=50000。接着n行，每行两个正整数h[i],w[i]，均不超过100000000。

【输出格式】输出文件books.out

输出一个正整数。

【输入输出样例1】

|  |  |
| --- | --- |
| books.in | books.out |
| 4  1 100  15 15  20 5  100 1 | 500 |

【说明】第一堆:100×1, 第二堆1×100, 第三堆20×5 和 15×15。每堆面积为100, 100, 300, 总共500。

**829.** **索道选址II**

(cable.cpp/c/pas)

【问题描述】

lester的老家有一个著名的旅游景点：大牛山。据说爬完就能成为C++大牛，所以游客不断。从山脚到山顶共有n个景点，排成一条直线。山顶为景点n，并且只有这一条登山路线orz。由于山太高，当地政府决定架设若干条索道站点。索道站点山上某个景点，并且到山顶（景点n）是必须有索道站点的。在景点i修建索道站点的花费为Ai。对于没有索道直达的景点，游客会乘坐高于该景点且最近的一个索道站点，然后从该站点向下走到目的地。步行游客会产生一定的不满意度，值等于索道站点与目的地编号之差。你可以认为想去每个景点的游客人数是相同的，例如索道站点在景点3，6，则去景点4需要先到景点6，然后向下走到景点4，不满意度为6-4=2。lester既不想花太多钱，又不愿意积累太多的不满意度。注意：游客出发的位置可以理解为在0号，而不是1号。他希望求一种折衷方案，使得花费与不满意度总和最小。

【输入格式】输入文件cable.in

第一行一个正整数n（<=100000）

第二行n个正整数Ai（<=1000000000）

【输出格式】输出文件cable.out

一个正整数，表示最优方案中，花费与不满意度的总和。

【输入输出样例1】

|  |  |
| --- | --- |
| cable.in | cable.out |
| 10  2 3 1 5 4 5 6 3 1 2 | 18 |

【说明】索道终点选在景点3，6，9，10（方案可能不唯一）

总花费=1+5+1+2=9

不满意度=(3-1)+(3-2)+(6-4)+(6-5)+(9-7)+(9-8)=9

总代价=18

**2621.清洗桌布1**

**(tablecloth.cpp/c/pas)**

【问题描述】

你是一家餐饮公司老板，为各种宴会上门提供美食，其中涉及到桌布的管理问题。接下去有连续T天的订单已经确认，其中第i天需要d[i]张干净桌布。注意：你一共只有m张桌布。每次使用过的桌布会变脏，如需再次使用必须送去清洗。共有2家清洗店，第一家清洗每张桌布收费c1元，需要n1天。第二家清洗每张桌布收费c2元，需要n2天。例如n1为1时，桌布可以第一天用完后当天就清洗第二天就可以使用了。你希望尽可能节省费用，请问为了用干净桌布完成所有订单要求，你最少需要为桌布付出多少清洗费用？如果无法提供干净桌布请输出-1。

【输入格式】输入文件**tablecloth**.in

输入第一行为正整数T，n1，n2，c1，c2，m，注意n1,n2<=T，c1,c2,<=60, m<=1000000。第二行为T个非负整数代表d[i]，均不超过50。

【输出格式】输出文件**tablecloth**.out

输出一个整数。

【输入输出样例1】

|  |  |
| --- | --- |
| **tablecloth**.in | **tablecloth**.out |
| 4 1 2 2 1 8  8  2  1  6 | 11 |

【说明】共4天，第1天用8张桌布，费用0元，使用后安排2张桌布1天洗完，6张桌布2天洗完，费用10元。第2天使用后安排一张桌布2天洗完，费用1元。第3天使用后不清洗。第4天使用后不清洗。

【输入输出样例2】

|  |  |
| --- | --- |
| **tablecloth**.in | **tablecloth**.out |
| 2 3 3 10 10 9  5  5 | -1 |

【数据规模】

50%数据，T<=500

100%数据，T<=100000

**2622.清洗桌布2**

**(cover.cpp/c/pas)**

【问题描述】

你是一家餐饮公司老板，为各种宴会上门提供美食，其中涉及到桌布的管理问题。接下去有连续T天的订单已经确认，其中第i天需要d[i]张干净桌布。当然你的桌布来源是上游供应商，每张桌布售价为p元。每次使用过的桌布会变脏，如需再次使用必须送去清洗。共有2家清洗店，第一家清洗每张桌布收费c1元，需要n1天。第二家清洗每张桌布收费c2元，需要n2天。例如n1为1时，桌布可以第一天用完后当天就清洗第二天就可以使用了。你希望尽可能节省费用，请问为了用干净桌布完成所有订单要求，你最少需要为桌布付出多少费用？

【输入格式】输入文件**cover**.in

输入第一行为正整数T，n1，n2，c1，c2，p，注意n1,n2<=T，c1,c2,p<=60。

后面T行，每行为1个非负整数代表d[i]，均不超过50。

【输出格式】输出文件**cover**.out

输出一个整数。

【输入输出样例1】

|  |  |
| --- | --- |
| **cover**.in | **cover**.out |
| 4 1 2 2 1 3  8  2  1  6 | 35 |

【说明】共4天，第1天买8张桌布，费用24元，使用后安排2张桌布1天洗完，6张桌布2天洗完，费用10元。第2天使用后安排一张桌布2天洗完，费用1元。第3天使用后不清洗。第4天使用后不清洗。

【输入输出样例2】

|  |  |
| --- | --- |
| **cover**.in | **cover**.out |
| 3 1 2 10 10 20  5  5  5 | 200 |

【数据规模】

50%数据，T<=500

100%数据，T<=100000