

NOIp Senior Day2

zhzh2001

题目名称	围栏	Quicksort Killer	花园改造
目录	corral	qkiller	landscape
可执行文件名	corral	qkiller	landscape
输入文件名	corral.in	qkiller.in	landscape.in
输出文件名	corral.out	qkiller.out	landscape.out
时间限制	1s	1s	1s
空间限制	512MB	512MB	512MB
测试点数量	20	10	20
测试点分数	5	10	5
比较方式	全文	SPJ	全文
部分分	无	无	无

提交源程序文件名

对于 C++ 语言	corral.cpp	qkiller.cpp	landscape.cpp
对于 C 语言	corral.c	qkiller.c	landscape.c
对于 Pascal 语言	corral.pas	qkiller.pas	landscape.pas

编译选项

对于 C++ 语言	-O2 -std=gnu++11	-O2 -std=gnu++11	-O2 -std=gnu++11
对于 C 语言	-O2	-O2	-O2
对于 Pascal 语言	-O2	-O2	-O2

注意事项：

1. 注意编译选项，避免未定义行为或编译错误。
2. 代码长度限制为 100KB。
3. 注意代码常数和 I/O 造成的效率影响。

1 围栏 (corral.c/cpp/pas)

1.1 题目描述

农夫约翰打算建造一个围栏来圈养他的奶牛。作为最挑剔的兽类，奶牛们要求这个围栏必须是正方形的，并且围栏里至少要有 C 个草场，来供应它们的食物。

约翰的土地上一共有 N 个草场 ($1 \leq C \leq N \leq 3,000$)，每个草场都在一块 1x1 的方格 内，坐标范围 $0 \dots 10^{18}$ 。如果有多个草场在同一个方格内，那么它们的坐标就会相同。

请帮助约翰计算围栏的最小边长。

1.2 输入格式 (corral.in)

第一行两个整数 C, N 。

接下来 N 行，每行两个整数，表示一个草场所在方格的坐标。

1.3 输出格式 (corral.out)

输出最小边长。

1.4 输入样例

```
3 4
1 2
2 1
4 1
5 2
```

1.5 输出样例

```
4
```

1.6 样例解释

围栏左上角坐标可以为 (1,1)，边长为 4，此时前三个草场都在围栏内。更小的边长不能满足要求。

1.7 数据范围

测试点	N	坐标范围
1	≤ 5	$0 \dots 10$
2		$0 \dots 10^{18}$
3	≤ 10	$0 \dots 1,000$
4	≤ 50	$0 \dots 50$
5		$0 \dots 200$
6		$0 \dots 10^{18}$
7	≤ 200	$0 \dots 200$
8		$0 \dots 5,000$
9		$0 \dots 10^{18}$
10	≤ 500	$0 \dots 500$
11		$0 \dots 5,000$
12		$0 \dots 10^{18}$
13	$\leq 1,000$	$0 \dots 5,000$
14		$0 \dots 10^{18}$
15	$\leq 2,000$	$0 \dots 500$
16		$0 \dots 10^{18}$
17	$\leq 3,000$	$0 \dots 200$
18		$0 \dots 1,000,000,000$
19		
20		$0 \dots 10^{18}$

2 Quicksort Killer(qkiller.c/cpp/pas)

2.1 题目描述

快速排序是一种广为使用的排序算法，其最好和平均时间复杂度均为 $\Theta(N \log N)$ ，但是最坏时间复杂度可达 $\Theta(N^2)$ 。为了简化问题，我们考虑给 N 个 32 位随机 整数排序 ($N = 100,000$)。

你的任务是，给定这 N 个整数，重新排列 这些整数，使得特定的快速排序实现运行时间超过 1s。

我们考虑一种常见的实现，来自Free Pascal 的示例代码：

```

1: procedure SORT( $A, l, r$ )                                ▷ 对  $A[l \dots r]$  排序
2:    $i \leftarrow l$ 
3:    $j \leftarrow r$ 
4:    $x \leftarrow A[F(l, r)]$                                 ▷  $x$  为基准值
5:   repeat
6:     while  $A[i] < x$  do
7:        $i \leftarrow i + 1$ 
8:     end while
9:     while  $x < A[j]$  do
10:       $j \leftarrow j - 1$ 
11:    end while
12:    if  $i \leq j$  then
13:      SWAP( $A[i], A[j]$ )
14:       $i \leftarrow i + 1$ 
15:       $j \leftarrow j - 1$ 
16:    end if
17:  until  $i > j$ 
18:  if  $l < j$  then
19:    SORT( $l, j$ )
20:  end if
21:  if  $i < r$  then
22:    SORT( $i, r$ )
23:  end if
24: end procedure

```

其中 $F(l, r)$ 返回一个 $[l, r]$ 之间的数，作为基准值的下标。这里，你需要考虑四种实现：

1. $F(l, r) = (l + r) / 2$
2. $F(l, r) = l$
3. $F(l, r) = r$
4. $F(l, r) = \text{random}(l, r)$

上述除法为整数除法， $\text{random}(l, r)$ 函数返回 $[l, r]$ 间的伪随机数。随机函数的工作原理如下：

在程序开始时，自动初始化随机数种子 $x_0 = \text{time}(\text{NULL})$ ， $\text{time}(\text{NULL})$ 函数同 C/C++ 中同名函数，即返回从 UTC 1970/01/01 00:00:00 开始到调用时经过的秒数。（不必考虑闰秒等问题）

$\text{random}(l, r)$ 函数第 i 次调用时 $x_i = x_{i-1} * 48,271 \bmod 2,147,483,647$ (即 `minstd_rand`)，返回值为 $x_i \bmod (r - l + 1) + l$ 。

2.2 输入格式 (qkiller.in)

第一行一个整数 $type (type \in [1, 4])$ ，表示 F 函数的实现。

如果 $type = 4$ ，接下来一行一个时间，格式为 `yyyy/mm/dd hh:mm:ss`，表示调用时间，范围从 UTC 1970/1/1 00:00:00 到 2020/12/31 23:59:59。

接下来一行 N 个整数，在 32 位有符号整数范围内。

2.3 输出格式 (qkiller.out)

输出 N 个符合要求的整数。

2.4 数据范围

测试点	$type$	整数范围
1	=1	$[-2^{31}, 2^{31})$
2		$[-10^8, 10^8]$
3	=2	$[-2^{31}, 2^{31})$
4		$[-1,000, 1,000]$
5	=3	$[-2^{31}, 2^{31})$
6		$[-1,000, 1,000]$
7	=4	$[-2^{31}, 2^{31})$
8		$[-10^8, 10^8]$
9		
10		

对于 100% 的数据，整数都在对应的范围内等概率随机生成。

3 花园改造 (landscape.c/cpp/pas)

3.1 题目描述

农夫约翰正在建造一个漂亮的花园，并且在这个工程中需要移动大量的泥土。

这个花园可以看作一个由 N 个花坛组成的序列 ($1 \leq N \leq 100,000$)，花坛 i 最初包含 A_i 个单位的泥土。农夫约翰想要改造这个花园，使得花坛 i 包含 B_i 个单位的泥土。 A_i, B_i 都是 $0 \dots 50$ 之间的整数。

为了改造花园，农夫约翰有几种选项：

- 花费 X 元购买一个单位泥土，并把它放在一个他选择的花坛中。
- 花费 Y 元移除他选择的花坛中一个单位泥土，并把它运走。
- 花费 $Z * |i - j|$ 元把一个单位泥土从花坛 i 运送到花坛 j 。

请计算完成改造计划的最小花费。

3.2 输入格式 (landscape.in)

第一行四个整数 N, X, Y, Z ($0 \leq X, Y \leq 10^8; 0 \leq Z \leq 1,000$)。

接下来 N 行，每行两个整数 A_i, B_i 。

3.3 输出格式 (landscape.out)

输出完成改造计划的最小花费。

3.4 输入样例

```
4 100 200 1
1 4
2 3
3 2
4 0
```

3.5 输出样例

```
210
```

3.6 样例解释

有一个单位的泥土必须被移除 (从第四个花坛)，花费 200 元。剩下的泥土可以花费 10 元移动 (三个单位泥土从第四个花坛到第一个，一个单位从第三个到第二个)。

3.7 数据范围

测试点	N	A_i, B_i
1	≤ 3	≤ 3
2	≤ 4	≤ 4
3	≤ 5	≤ 5
4	≤ 10	
5		
6	≤ 20	≤ 10
7	≤ 30	
8	≤ 50	
9	≤ 100	
10		
11		≤ 20
12		≤ 30
13	≤ 500	≤ 50
14	$\leq 1,000$	
15	$\leq 2,000$	
16	$\leq 5,000$	
17	$\leq 20,000$	
18	$\leq 50,000$	
19	$\leq 100,000$	
20		