

# CSP 非专业级软件能力认证

## 提高级

### CSP-S 2019 模拟赛

#### 第二试

时间：2019 年 11 月 14 日 08:30 ~ 12:00

题目名称	a	b	c
题目类型	传统型	传统型	交互型
目录	a	b	c
可执行文件名	a	b	c
输入文件名	a.in	b.in	c.in
输出文件名	a.out	b.out	c.out
每个测试点时限	1.0 秒	1.0 秒	3.0 秒
内存限制	512 MB	512 MB	512 MB
子任务数目	4	5	5
测试点是否等分	否	否	否

提交源程序文件名

对于 C++ 语言	a.cpp	b.cpp	c.cpp
对于 C 语言	a.c	b.c	c.c
对于 Pascal 语言	a.pas	b.pas	c.pas

编译选项

对于 C++ 语言	-O2 -lm -std=c++11
对于 C 语言	-O2 -lm
对于 Pascal 语言	-O2

## a (a)

## 【题目背景】

艾登·皮尔斯又跳河了。

## 【题目描述】

在一个竖直的二维平面上， $x$  轴为水平轴， $y$  轴为竖直轴。其中  $x$  轴是地面，并且区间  $[l, r]$  被水覆盖。

艾登从  $(0, h)$  开始自由落体。（不考虑空气阻力）

因为你收到了来自远房亲戚 Madeline 的祝福，所以艾登可以在任意时刻使用一次向八个方向之一的冲刺，使得速度的改变量为这个方向上长为  $t$  的向量。（八个方向：上、下、左、右、左上、右上、左下、右下）

求最小落入水中的速度。如果不能入水，则输出 "-1"。

本题中  $g = 9.8 \text{ m/s}^2$  ! 本题中  $g = 9.8 \text{ m/s}^2$  ! 本题中  $g = 9.8 \text{ m/s}^2$  !

## 【输入格式】

从文件 **a.in** 中读入数据。

一行四个整数  $l, r, h, t$ 。

## 【输出格式】

输出到文件 **a.out** 中。

一个浮点数或 "-1"。

如果答案不为 "-1"，保留两位小数（数据保证真实答案  $\times 100$  的小数部分与 0.5 之差大于  $10^{-2}$ ）

否则，输出字符串 "-1"。

## 【子任务】

总限制：输入的所有数字都为绝对值不大于  $10^4$  的整数。 $h > 0$ 。

测试集编号	特殊限制	分值
1	$t = 0$	30
2	$l = r = 0$	20
3	$l = r$	30
4		20

b (b)

【题目背景】

三人同心，其利断金。

【题目描述】

有三人在一个水平的二维平面上，抽象为三个相同的点。  
已知他们初始坐标的可重集，和目标坐标的可重集。  
一次移动可以将一个点变为其关于另一个点的对称点。  
判断是否可行。

【输入格式】

从文件 *b.in* 中读入数据。  
六行，每行两个整数描述一个坐标。  
前三个坐标为初始坐标。  
后三个坐标为目标坐标。

【输出格式】

输出到文件 *b.out* 中。  
"Yes" 或 "No"

【子任务】

总限制：输入的所有数字都为绝对值不大于  $10^9$  的整数。

测试集编号	特殊限制	分值
1	初始三个点 $y = 0$	30
2	初始三个点和目标三个点都构成等腰直角三角形且全等	20
3	初始三个点和目标三个点都构成三角形且全等	20
4	初始三个点构成三角形	10
5	无特殊限制	20

## c (c)

这是一道交互题。

**【题目背景】**

艾登·皮尔斯又被通缉了。

**【题目描述】**

有  $n$  座 ctOS 塔，每座塔有一个病毒值，初始全都为 0。

进行  $m$  次操作。

- 操作 1，输入  $i, v$ ，表示艾登成功将第  $i$  座 ctOS 塔的病毒值增加  $v$ 。
- 操作 2，输入  $S, b$ ，表示城市新增了一位警员，当集合  $S$  内的 ctOS 塔的病毒值之和小于  $b$  的时候，这位警员会一直追击艾登，直到不满足条件。

为了方便，ctOs 塔从 0 开始标号（即  $i \in [0, n)$ ）。

$S$  是一个小于  $2^n$  的自然数，其二进制下第  $t$  位表示集合内是否有  $t$ 。

每次操作后输出正在追击艾登的警员数量。

**强制在线 强制在线 强制在线**（为了扫盲再来一遍）**强制在线 强制在线 强制在线**

为了实现强制在线，本题采用交互题的形式。

你需要实现两个函数：

```
void init(int n,int m){ srand(time(0)^clock()); }
int doit(int opttype,int input1,int input2){ return rand(); }
```

（以上是一份存在一定几率可以 AC 此题的实例代码）

**【输入格式】**

交互库会先调用一次 `init`，告诉你这组数据的  $n$  和  $m$ 。

之后会调用  $m$  次 `doit`，传入的三个数描述这次操作。

**【输出格式】**

每次调用 `doit` 时，应返回一个整数表示这次操作结束后的答案。

**【子任务】**

总限制：所有输入的数字都为不大于  $10^9 + 7$  的自然数。 $0 \leq i < n$ 。 $0 \leq S < 2^n$ 。

测试集编号	n=	m=	其他限制	分值
1	18	1001		10
2	4	200002		10
3	18	200003	集合大小 $\leq 6$ 且集合总大小 $\leq 30000$	30
4	18	200004	可离线	30
5	18	200005		20

**请注意：**如果没有强制在线 ( $m = 200004$ )，那么你可以在任意时刻从一个名字叫 offline 的 `vector<int>` 中预言之后的  $3 \times m$  个数，就像以下代码一样。

```
void init(int n,int m){
    for (int i=0;i<3*m;++i)
        cerr<<offline[i]<<endl;
    // 第 i 个数为第 i/3+1 次 doit 的第 i%3+1 个参数。
}
```

如果强制在线 ( $m \neq 200004$ )，则不要访问 offline，后果自负。