

# 华为OD机试 - 经典屏保（Java & JS & Python）

原创

伏城之外

已于 2023-06-02 22:08:15 修改

1055

收藏 8

版权

分类专栏：

华为OD机试AB（Java & JS & Python）

文章标签：

算法

华为机试

Java

JavaScript

Python

OD

华为OD机试AB（Ja... 同时被 2 个专栏收录

该专栏为热销专栏榜 第2名

¥59.90

¥99.00

3382 订阅

371 篇文章

已订阅

## 题目描述

DVD机在视频输出时，为了保护电视显像管，在待机状态会显示“屏保动画”，如下图所示，DVD Logo在屏幕内来回运动，碰到边缘会反弹。



请根据如下要求，实现屏保Logo坐标的计算算法。

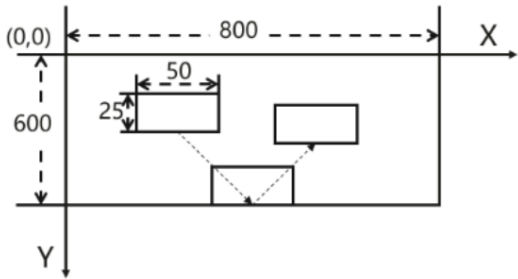
1.

屏幕是一个800\*600像素的矩形，规定屏幕的左上角点坐标原点，沿横边向右方向为X轴，沿竖边向下方向为Y轴
2.

Logo是一个50\*25像素的矩形，初始状态下，左上角点坐标记做(x, y)，它在X和Y方向上均以1像素/秒的速度开始运动
3.

遇到屏幕四个边缘后，会发生镜面反弹，即以45°碰撞边缘，再改变方向以45°弹出
4.

当Logo和四个角碰撞时，两个边缘同时反弹的效果是Logo会原路返回



请编码实现，t秒后Logo左上角点的坐标。

输入描述

输入3个数字，以空格分隔：

```
x y t
```

第一个数字表示Logo左上角点的初始X坐标；  
第二个数字表示Logo左上角点的初始Y坐标；  
第三个数字表示时间 t，题目要求即求 t 秒后Logo左上角点的位置。

输出描述

输出2个数字，以空格分隔：

```
x y
```

第一个数字表示 t 秒后，Logo左上角点的X坐标  
第二个数字表示 t 秒后，Logo左上角点的Y坐标

备注

所有用例均保证：

- 输入的x和y坐标会保证整个Logo都在屏幕范围内，Logo不会出画；
- 所有输入数据都是合法的数值，且不会出现负数；
- t 的最大值为100000。

用例



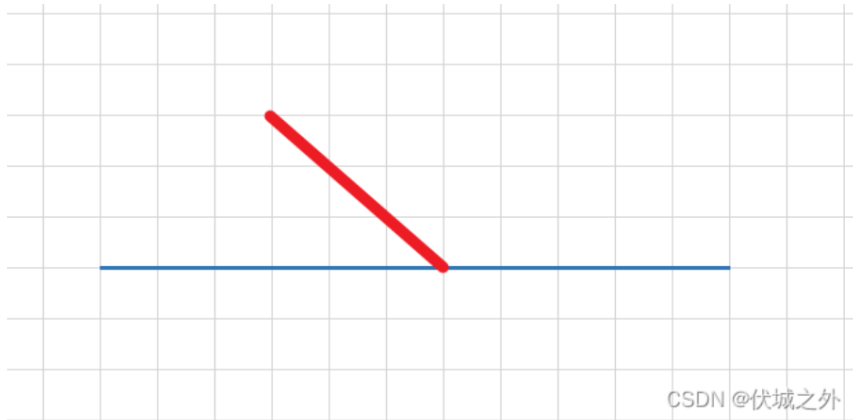
输入	0 0 10
输出	10 10
说明	输入样例表示Logo初始位置在屏幕的左上角点，10s后，Logo在X和Y方向都移动了10像素，因此输出10 10。

输入	500 570 10
输出	510 570
说明	输入样例表示初始状态下，Logo的下边缘再有5像素就碰到屏幕下边缘了，5s后，会与屏幕碰撞，碰撞后，斜向45°弹出，又经过5s后，Logo与起始位置相比，水平移动了10像素，垂直方向回到了原来的高度。

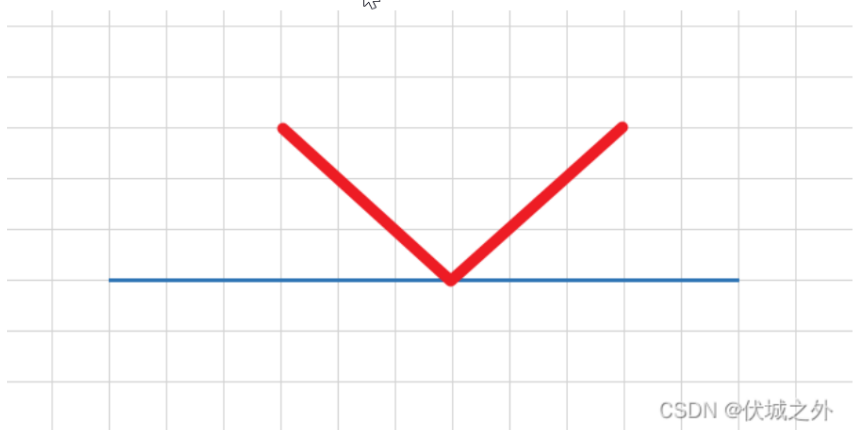
## 题目解析

这道题其实可以转化为一个简单的物理常识题，或者看出一个数学问题。

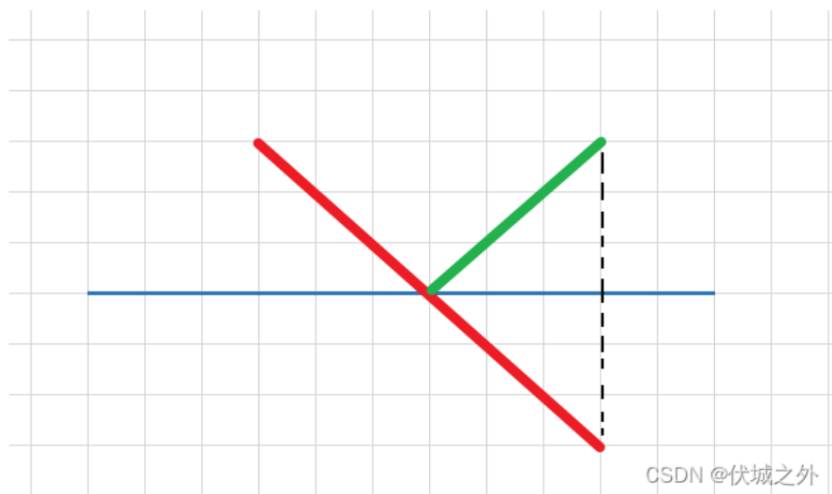
举个例子，有一束红外线，和一个水平放置的镜面，假设红外线以和水平镜面 $45^\circ$ 夹角的位置射向镜面，问红外线会如何反射，如下图



此时有两种策略，第一个策略是以红外线和镜面的交点开始，画一条与红外线呈 $90^\circ$ 夹角的线，这条线就是红外线的反射线



还有一个策略就是，让红外线穿透镜面后，沿着镜面对称反转



而本题就可以利用策略二思路解题。

得到 $x, y, t$ 后，我们可以直接让

- $x += t$
- $y += t$

这其实就相当于让红外线穿透镜面（如果 $t$ 足够大）

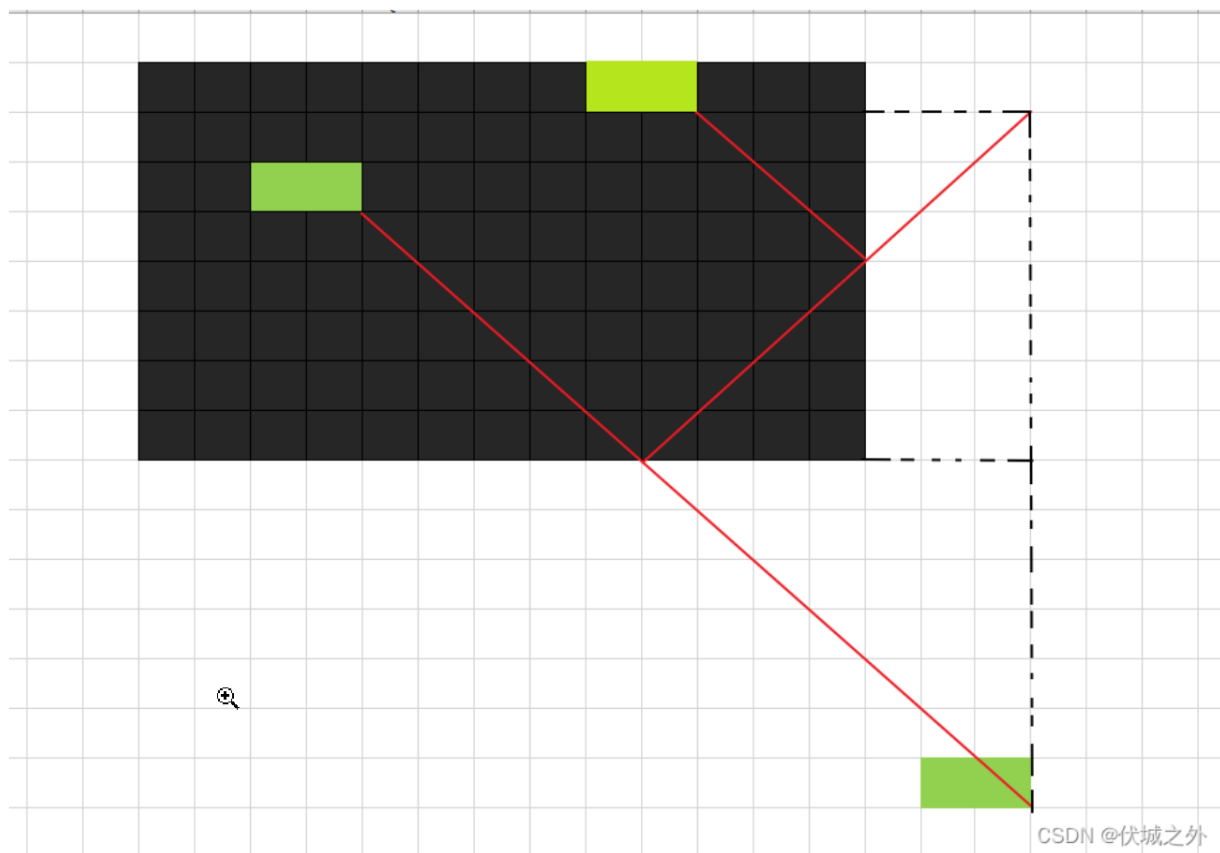
之后，我们检查logo是否越界

- 若  $x + 50 > 800$ ，其中 $x + 50$ 是logo的右下角横坐标，则说明越界，此时我们需要沿着 $x=800$ 轴进行对称反转，反转后 $x+50$ 坐标变为了  $800 - (x+50 - 800)$ ，进而得到反转后  $x = 800 - (x+50 - 800) - 50$
- 若  $y + 25 > 600$ ，其中 $y + 25$ 是logo的右下角纵坐标，则说明越界，此时我们 $y = 600$ 轴进行对称反转，反转后 $y + 25$ 坐标变为了  $600 - (y+25 - 600)$ ，进而得到反转后  $y = 600 - (y+25 - 600) - 25$

当然除了上面logo越界情况，还有两个越界情况，如下：

- 若  $x < 0$ ，其中 $x$ 是logo的左上角横坐标，则说明越界，此时我们需要沿着 $x=0$ 轴进行对称反转，即反转后  $x = -x$
- 若  $y < 0$ ，其中 $y$ 是logo的左上角纵坐标，则说明越界，此时我们需要沿着 $y=0$ 轴进行对称反转，即反转后  $y = -y$

上面逻辑，大家可以对照下图思考一下



如果  $t$  足够大，则会进行多次越界反转逻辑，因此我们需要while循环判断反转后的  $x$ ,  $y$ ,  $x+50$ ,  $y+25$  是否越界，如果越界则继续反转逻辑，否则就退出循环。

最后的  $x, y$  一定是不越界的，即最终坐标位置，此时可以作为题解返回。

```
1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         System.out.println(getResult(sc.nextInt(), sc.nextInt(), sc.nextInt()));
7     }
8
9     public static String getResult(int x, int y, int t) {
10         x += t;
11         y += t;
12
13         while (y + 25 > 600 || y < 0 || x + 50 > 800 || x < 0) {
14             if (y + 25 > 600) {
15                 // y = 600 - (y + 25 - 600) - 25;
16                 y = 1150 - y;
```

```
17     }
18
19     if (x + 50 > 800) {
20         //      x = 800 - (x + 50 - 800) - 50;
21         x = 1500 - x;
22     }
23
24     if (y < 0) {
25         y = -y;
26     }
27
28     if (x < 0) {
29         x = -x;
30     }
31 }
32
33 return x + " " + y;
34 }
35 }
```