

# 缀点成线

## 题目描述

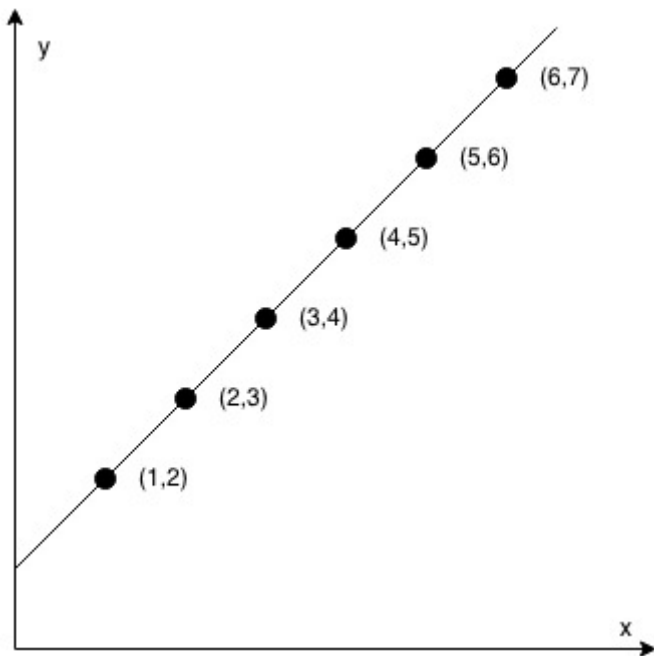
在一个 XY 坐标系中有一些点，我们用数组 `coordinates` 来分别记录它们的坐标，其中 `coordinates[i] = [x, y]` 表示横坐标为 `x`、纵坐标为 `y` 的点。

请你来判断，这些点是否在该坐标系中属于同一条直线上，是则返回 `true`，否则请返回 `false`。

示例 1:

输入: `coordinates = [[1,2],[2,3],[3,4],[4,5],[5,6],[6,7]]`

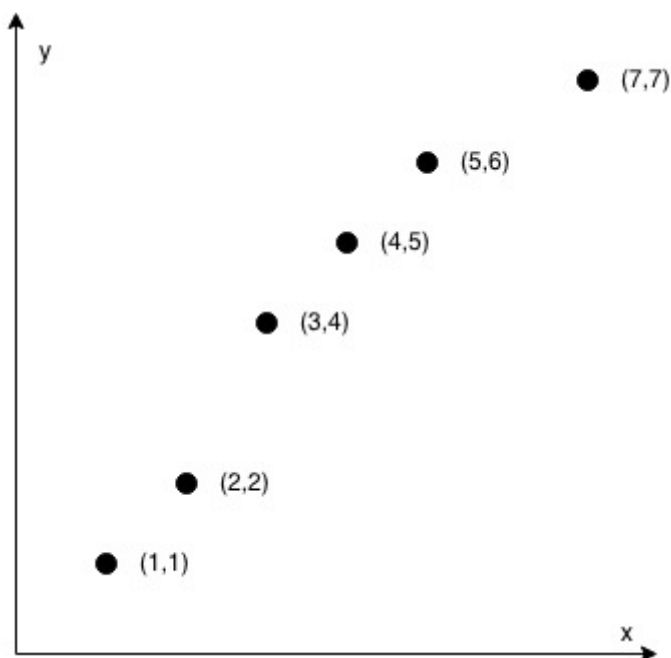
输出: `true`



示例 2:

输入: `coordinates = [[1,1],[2,2],[3,4],[4,5],[5,6],[7,7]]`

输出: `false`



提示:

- $2 \leq \text{coordinates.length} \leq 1000$
- $\text{coordinates}[i].\text{length} == 2$
- $-10^4 \leq \text{coordinates}[i][0], \text{coordinates}[i][1] \leq 10^4$
- `coordinates` 中不含重复的点

## 解析

- 可以使用斜率来进行计算，定两个定点，分别为第一个点和最后一个点，如果第三个点在这个直线这个，那么斜率一定相同。
- 第一个点为 (X1, Y1)
- 最后一个点为 (x2, y2)
- 需要计算的点为 (x, y)
- 三点在一条直线上一定满足

$$(x - x1) / (y - y1) = (x - x2) / (y - y2)$$

// 为了防止除零出现，变换成乘法

$$(x - x2) * (y - y1) = (x - x1) * (y - y2)$$

## 代码实现

C++

```

class Solution {
public:
    bool checkStraightLine(vector<vector<int>>& coordinates) {
        if (coordinates.size() == 0)
            return 0;
        int x_count = coordinates.size();
        for (int i = 1; i < x_count - 1; i++)
        {
            if ( (coordinates[i][0] - coordinates[x_count - 1][0]) * (coordinates[i][1] - coordi
            != \
            (coordinates[i][0] - coordinates[0][0]) * (coordinates[i][1] - coordinates[x_count -
            )
            return false;
        }
        return true;
    }
};

```