Debug Lab 1

Intro

在本Lab中，你需要修复5个有问题的程序中的BUG，使得程序能正确执行。

5个程序分别存放在`P1`，`P2`，`P3`，`P4`，`P5`文件夹下。每个文件夹下会有一些头文件，一个`main.cpp`以及一个`px.cpp`（`x`代表一个数字，如果`x`是1就是`p1.cpp`）。请注意你只能修改`px.cpp`以修复BUG。在对代码进行评测时我们会使用原本的头文件以及`main.cpp`进行编译

为了防止你通过重写代码来解决问题，我们会使用Unix Shell的`diff`工具来比较你上传的`px.cpp`和原本的`px.cpp`。每一题都有各自的`diff`输出行数的限制。这个限制通常留有足够的余量，因此你不必对只改了几行就修复了BUG感到疑惑。

`diff`工具的效果如上图所示，`diff a.txt b.txt`一共输出了8行。为了不干扰`diff`，请尽量不要在代码中添加无意义的空格和换行

如果是windows用户，可以使用如下工具进行替代来检查自己做了哪些修改（比如有没有加空格）。不过要注意`fc.exe`的输出包含一些空行，所以输出行数通常比`diff`多

```
fc.exe .\p3.cpp .\p3_answer.cpp | Measure-Object -line
```

P1：Easy Link

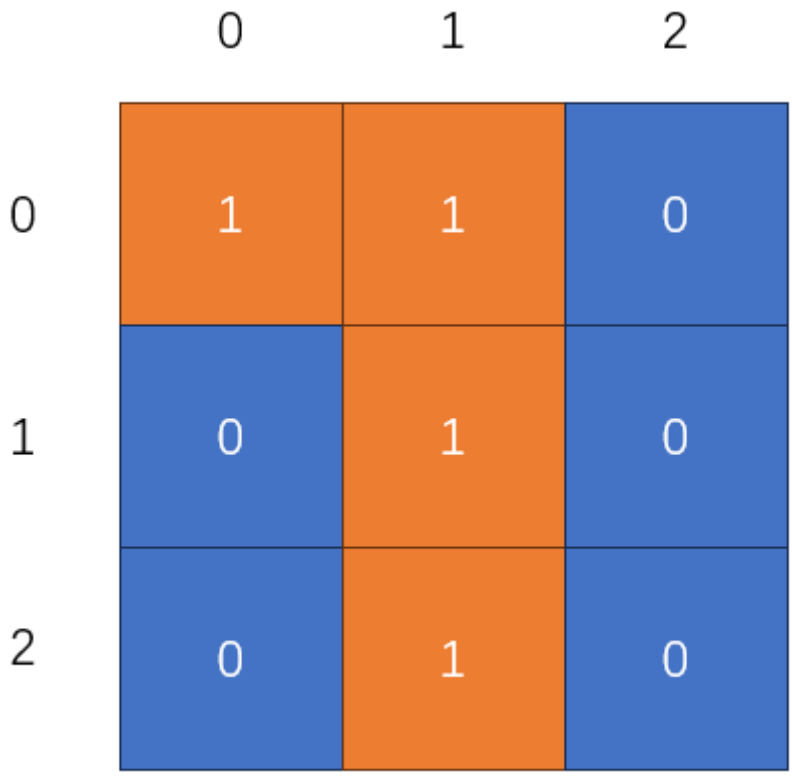
问题描述

助教最近在研究Qlink的实现。为了做些练习，助教写了一个小程序，程序想完成如下目标：

给定一个n * m的矩阵，矩阵由整数0或1组成，0表示可以通行，1表示无法通行。同时给定一个起点(x_{start}, y_{start})和终点(x_{end}, y_{end})，求能否从起点开始，在只经过可以通行的矩阵格子的情况下，到达终点（假设起点和终点都可以通行）。

如果能到达则输出1，否则输出0。

样例



样例一：

- 输入：

```
3 3 2 0 0 2
1 1 0
0 1 0
0 1 0
```

- 输出：

```
0
```

- 分析：矩阵如上，$(x_{start}, y_{start}) = (2, 0)$，$(x_{end}, y_{end}) = (0, 2)$。$(2, 0)$代表左下角的点，$(0, 2)$代表右上角的点。显然它们被橙色的无法通行的格子隔开了

样例二：

- 输入：

```
3 3 2 2 0 2
1 1 0
0 1 0
0 1 0
```

- 输出：

```
1
```

- 解释：矩阵如上，$(x_{start}, y_{start}) = (2, 2)$，$(x_{end}, y_{end}) = (0, 2)$。$(2, 2)$代表右下角的点，$(0, 2)$代表右上角的点。显然没有东西隔开它们

你的任务

助教写了一个程序，放在P1文件夹下，助教发现这个程序总是挂掉，请你帮助教解决这个问题。

Hint：程序挂掉会不会是因为内存访问越界？

经过你修改的`p1.cpp`与原本的`p1.cpp`的`diff`输出应少于**20**行

P2：汉诺塔

问题描述

同Lab3 - 汉诺塔

样例

同Lab3 - 汉诺塔

你的任务

助教在研究汉诺伊塔这个lab的时候自己写了一个程序，但这个程序总是报错。助教很苦恼，请你帮助教修复这个问题。

Hint：`stack.h`中包含了模板栈的实现，里面有一些`assert`断言，研究一下断言会在什么时候被触发？

Hint：除了断言的问题之外还有一些低级错误。

经过你修改的`p2.cpp`与原本的`p2.cpp`的`diff`输出应少于**30**行

P3：数独

本问题来自Leetcode 37. 解数独

问题描述

编写一个程序，通过填充空格来解决数独问题。

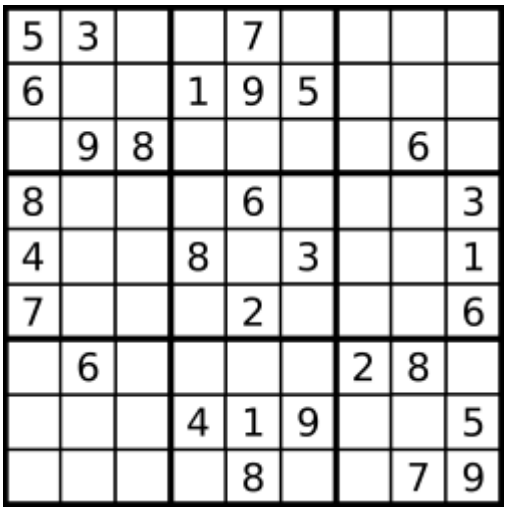
数独的解法需 **遵循如下规则**：

1. 数字 `1-9` 在每一行只能出现一次。
2. 数字 `1-9` 在每一列只能出现一次。
3. 数字 `1-9` 在每一个以粗实线分隔的 `3x3` 宫内只能出现一次。（请参考示例图）

~~数独部分空格内已填入了数字，空白格用 `'.'` 表示。~~

为了方便处理输入，本题的 `main.cpp` 中使用 `0` 来代表空白格

样例



5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

输入：

```
5 3 0 0 7 0 0 0 0
6 0 0 1 9 5 0 0 0
0 9 8 0 0 0 0 6 0
8 0 0 0 6 0 0 0 3
4 0 0 8 0 3 0 0 1
7 0 0 0 2 0 0 0 6
0 6 0 0 0 0 2 8 0
0 0 0 4 1 9 0 0 5
0 0 0 0 8 0 0 7 9
```

输出：

```
5 3 4 6 7 8 9 1 2
6 7 2 1 9 5 3 4 8
1 9 8 3 4 2 5 6 7
8 5 9 7 6 1 4 2 3
4 2 6 8 5 3 7 9 1
7 1 3 9 2 4 8 5 6
9 6 1 5 3 7 2 8 4
2 8 7 4 1 9 6 3 5
3 4 5 2 8 6 1 7 9
```

解释：输入的数独如上图所示，唯一有效的解决方案如下所示：

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

你的任务

助教写了一个程序来解数独，但是这个程序完全无法解出数独，请你帮助助教把程序改对。

Hint: 你可以认为注释传达了助教原本的意图，所有的注释都是对的。

经过你修改的`p3.cpp`与原本的`p3.cpp`的`diff`输出应少于**30**行

P4：铺瓷砖

问题来自Leetcode 1240. 铺瓷砖

问题描述

你是一位施工队的工长，根据设计师的要求准备为一套设计风格独特的房子进行室内装修。

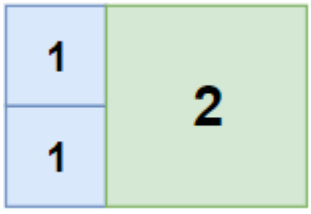
房子的客厅大小为 `n` x `m`，为保持极简的风格，需要使用尽可能少的 **正方形** 瓷砖来铺盖地面。

假设正方形瓷砖的规格不限，边长都是整数。

请你帮设计师计算一下，最少需要用到多少块方形瓷砖？

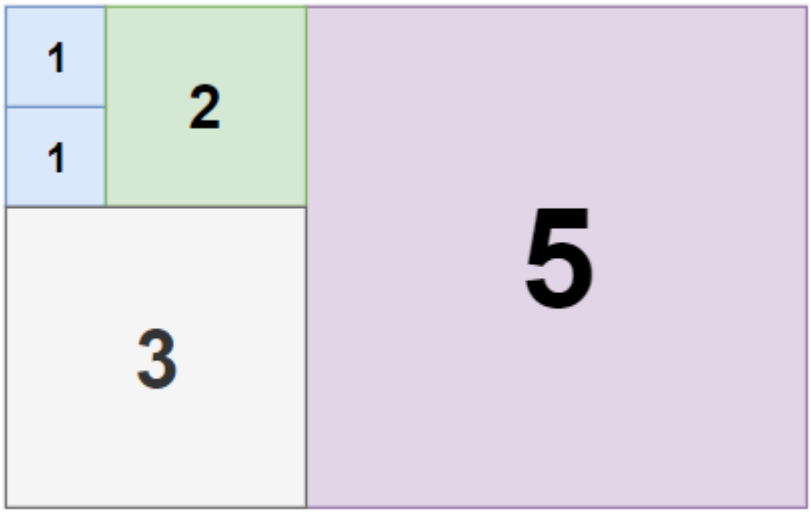
样例

示例 1：



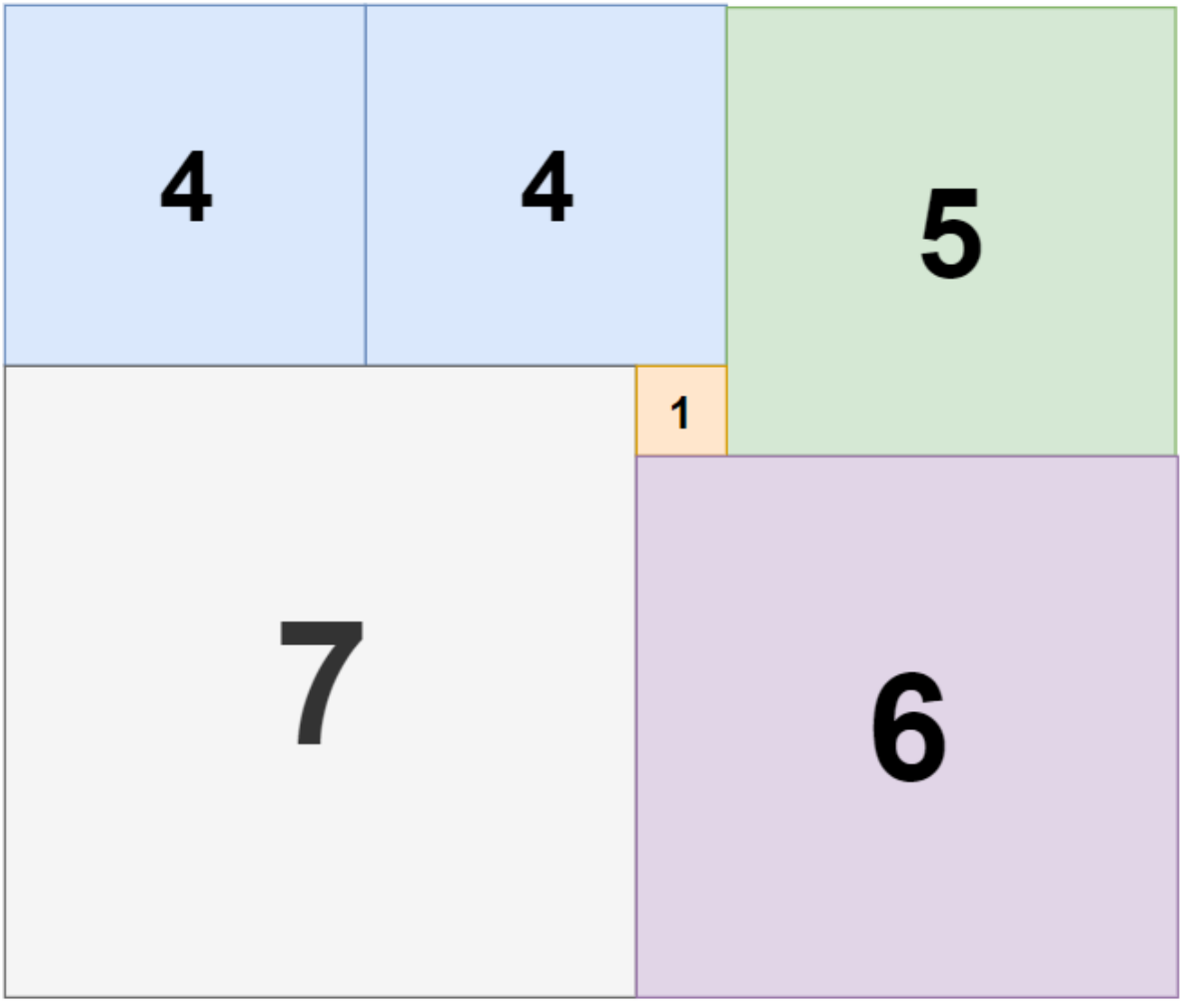

```
输入：2 3
输出：3
解释：3 块地砖就可以铺满卧室。
      2 块 1x1 地砖
      1 块 2x2 地砖
```

示例 2：




```
输入：5 8
输出：5
```

示例 3：




```
输入：11 13
输出：6
```

提示：

- `1 <= n <= 50`
- `1 <= m <= 50`

- 是的，数据范围比Leetcode上的题目要大

你的任务

助教想了想，决定使用搜索算法解决这个问题。但是怎么都写不对，请你帮助教把程序改对。

Hint: 如果无法修复问题，可以尝试检查`dfs`中每一个变量的行为是否符合预期

经过你修改的`p4.cpp`与原本的`p4.cpp`的`diff`输出应少于**60**行

P5：C++是最好的语言吗

问题描述

助教最近学会了C++，觉得C++是世界上最好的语言，并决定写点程序练练手。助教的程序想要依次完成三个任务：

1. 创建三个栈，编号为0， 1， 2
2. 反复向栈中填入数字
3. 反复弹出栈中元素直到栈为空

程序输入如下

1. 一个整数n，代表接下来会有n行输入
2. n行输入，每一行由两个数字i 和x，代表向编号为i的栈填入数字x

预期输出如下：

1. 三行输出，每行为对编号为i的栈不断进行出栈的结果

样例

输入：

```
5
0 2
0 3
1 4
2 5
1 1
```

输出：

```
stack 0: 3 2
stack 1: 1 4
stack 2: 5
```

解释：

对编号为0的栈依次入栈2和3，所以出栈的结果是3和2。其他栈同理

你的任务

助教写了一个程序，但发现程序总是无法正确输出，请你将程序改对。

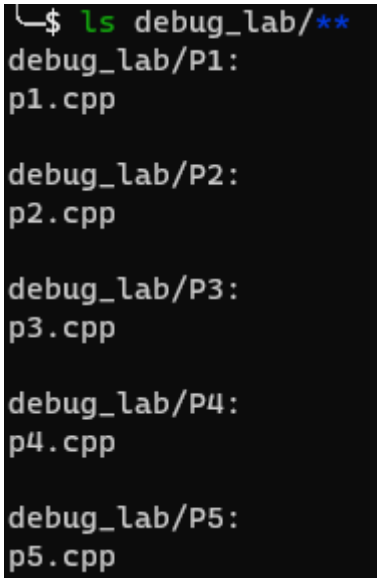
Hint: 如果你对C++不是特别熟悉，可以通过打印log，单步执行等方式来确认`p5.cpp`定义的诸多函数中究竟有哪几个以怎样的顺序被执行了

Hint: 如果你无法理解为什么程序为这么执行，可以自行查阅关于C++的左值、右值、临时量、生命周期等概念的资料

经过你修改的`p5.cpp`与原本的`p5.cpp`的`diff`输出应少于**30**行

提交格式

你需要上传一个名为`debug_lab.7z`的压缩包，压缩包解压出来应是一个名为`debug_lab`的文件夹，其中包含`P1`，`P2`，`P3`，`P4`，`P5`5个文件夹，每个文件夹中含有你修改过的`px.cpp`文件。如下图所示



你可以在`debug_lab`文件夹中放置其它文件，但是这些文件不会被参与评测。