学习文档：

对于github没有过深了解，目前仅学会上传文件，对操作界面仍然欠缺掌握。

针对1-1之前的内容，我通过慕课网的课程和c程序设计书籍进行学习



在c语言的基础类型方面，已掌握整型/布尔/浮点/数组/字符

对于void空型了解但未运用，个人认为可归于函数领域。

指针上因为在本次测试题难度并不需要深度掌握这两方面知识，并未深度掌握，仅仅初步了解，

结构类型已掌握数组，尚未学习结构体等应用较少类型，仅初步了解，由于缺乏具体应用情景未充分理解其应用。

字符方面c语言好像与其他语言有些不同，没有string 定义的字符串而要用char【】代替



程序控制语句方面掌握 if for while 循环判断赋值计算输入输出等内容，对于switch等应用较少语句还不能熟练运用 ，但初步了解形式。

函数方面对函数使用/迭代递归算法已了解但未进行过使用，不算熟练掌握。

多位数组一能较熟练运用，文件读写在学习1-2时部分掌握。

一些难点与克服：

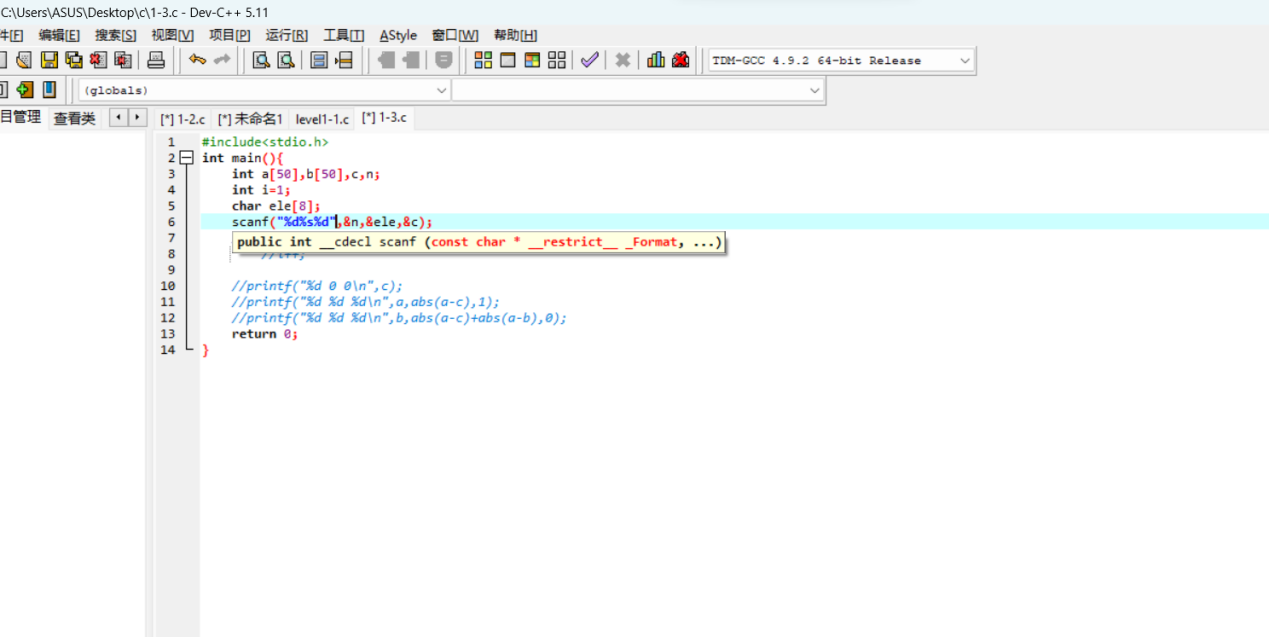
学习文件输入时存在一定困难，书本的阐述较为抽象，通过stfw 了解源代码

目前可以做到在输入文件上进行输出。1-2已提交。

使用fsanf格式后调试存在一定困难，因此认为应先采取手动输入格式，调试无误后再转为文件。

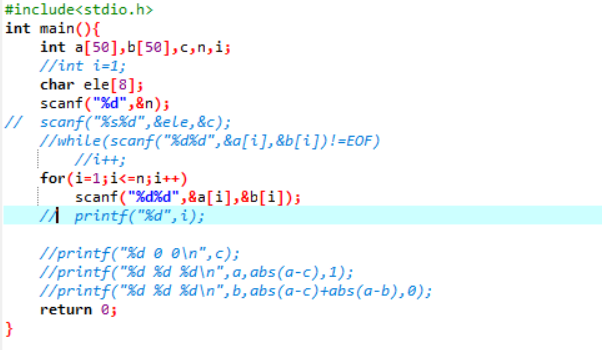
对于1-1 ：

掌握输入输出后不存在难点，唯一需要注意的是要引入abs函数。



对于1-3：已提交。

此题耗时较久。第一个难点存在于输入方面，未知输入数量时变量的读写。通过学习知道可以通过scanf()！=eof方式检测是否仍有输入，但尽可用于文件输入，手动输入时会发现无法结束输入，循环会一直进行下去。当时未注意到群里关于输入q结束输入的方案，因此采用在输入时引入等待人数n以提示循环次数的方法。



（对手动输入方式进行调试

第二个难点在于如何对已进入电梯的人员/已出电梯的人员进行相应标记。

我首先想到的时布尔类型判断，需要引入布尔数组，可以有效解决问题但程序不够简洁。

因此采取将已进入电梯人员的初始楼层a[i]改为0的策略，使其不会被重复计数并可以及时出电梯退出循环以确定电梯内人数，同时出电梯后再次将a[i]改为-1进行第二次标记。

第三个难点在于算法的选用，

fCfs算法在同时按电梯情境下不适用；

Sstf 算法如果仅考虑到达乘客所在楼层时间很好实现代码，唯一难点在于满载时面对同一层乘客应该先搭乘谁。可以采用定义函数递归进行时间计算比较。

我采用的时scan算法，主要难点在于要使得电梯循环上下运动需要对for循环做出一些改进，具体实现如下：



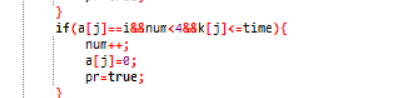
不使用i++而使用i+=con,同时利用所在楼层与con的状态进行判断是否调整con的数值，在1与-1之间进行切换，以改变电梯上下行状态。

与此同时，2-2的上下键问题也可以通过a[i]与b[i]的大小及con的取值以进行处理。

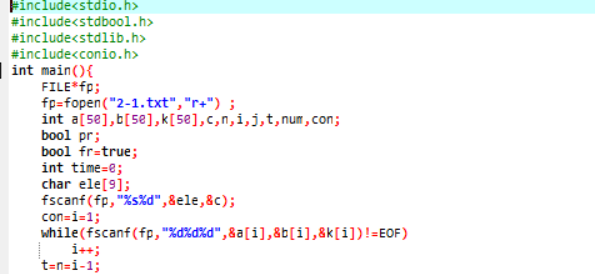
最后一个难点在于输出，如上图我通过对电梯停靠事件以布尔类型标记处理。

2-1：

2-1与1-3基本相同，以scan算法处理唯一区别在于引入对time的判断，引入时间数组c[i]若其小于time便可进入电梯。如下：



同时由于此题目要求了输入格式，不能再引入n来判断组数，因此我将程序改为文件读写，使用eof判断输入是否结束。同时测算出n的大小。如下：



Ps：因为1-3用的scan直接延续下来，如果要look加个判断即可

2-2：遭遇难题：2台电梯应该如何处理？不能再使用往复进行的for循环而要着重考虑时间的因素。