**学习记录**

0-1

按照提示和网上查找的教程，我下载了git，并在GitHub官网上注册了自己的账号，将git与账号绑定，建立了自己的仓库dian，成功将远程仓库链接提交。

0-2

由于先前并未接触过c语言，我开始通过看书、找网课等方式学习c语言。我明白了头文件的作用，了解了c语言编写的各种格式，懂得了编译器将c语言转化为目标程序，再生成可执行程序的过程。我学会了一些基础的函数语句，知道了如何输入输出数据；学会了引入数组，向数组输入数据；学会了两种选择语句、三种循环语句，并结合例题充分感受到循环语句提升效率之大。此时我以及可以根据所需画出一些简单的算法框图，并能将其编写为程序。

1-1

由于只有一个人乘坐电梯，只需要进行一些加减运算即可，主要是按照要求的格式进行输入输出即可。

1-2

文件读写并未学明白，我只能按照书上的格式来做，大概理解了他是定义了一个指向文件型数据的指针变量，并将数据文件的位置信息给他，至于如何读取数据文件中指定的几个数据，我还不清楚。

1-3

这里由于有多个乘客，电梯以怎样的优先级接送乘客就成了问题。我首先采用了电梯优先就近接人，之道满载，再开始就近送人，如果在接人（送人）的沿途有可以送（接）的人，就顺便执行，可是这样的效率并不高，电梯来回上下调动，容易出现在一个楼层附近反复上下的情况，且无法及时顾及较远的服务请求，加之在编写上遇到了很多我没想出解决办法的问题，我决定换一个思路。经过我对作业调度算法的初步了解学习，我选用scan算法，让电梯直上直下，尽可能照顾到每一个乘客，并且放弃了之前每接送一个人进行一次时间计算的时间算法，改用每层时间t++的方式，这样更适合scan算法，每走一层进行一次判断是否需要接送人，并使时间加一。确定思路后，我在编写上也遇到了不少困难。首先就是对数组数据的处理，刚开始（对应的是我的第一种想法）我想让所有乘客的所在楼层和目标楼层分别按大小顺序排序，让电梯找到自己所在楼层在其中的位置，但是这样就无法将每一个乘客的所在和目标楼层对应起来，数据混乱了；并且回头来想，我这里犯了一个惯性思维的错误，因为在人看来，如果有一串有大小顺序的数字，那么再来任何一个数字（实际上就是电梯所在楼层）就可以很快很准确的判断出距离它最近的数组元素是哪一个；但是对计算机而言，这样还不如计算每一个数组数据和电梯所在楼层的差值，然后比较大小，只需要循环结构即可找出距离最近楼层，并且没有改变数组数据的顺序，可以用两组数组，其中同一下标的数组元素为同一个人的所在和目标楼层，十分方便。之后又遇到了如何排除已经用过的数据的问题，例如a[2]表示第三个乘客的所在楼层，如果电梯已经接上他，但是他的数据仍保留在a[2]中，则电梯在再一次来到此楼层后又会进行一次接客，就会发生错误。我开始的思路是，每用过一个数组元素，就让后面的数据依次前移一个位置，将旧数据挤掉，但是这样就会发生最后一个数据无法抹去、出现两次的问题；之后我改用改变用过数组元素的数据的方法，让他变成一个很大的值，并在循环结构中加一个判断，如果数据大于10楼则忽略他，不进行停留。如此处理过后，1-3的scan版本也大体完成，不过我通过学习知道还有比scan更高效的look算法，即可以及时调整方向的算法，我进一步对1-3进行了改进。我的思路是在每一次接送人之后进行一次判断，上行时判断电梯所在楼层是否比所有乘客所在楼层和电梯内乘客目标楼层都大，下行时判断电梯所在楼层是否为最小，这样确实可行，但是由于每次执行接送人任务后都要判断很多，导致代码极度冗长，这方面我暂时没有想到好的解决方法，但总归1-3的look版最终也完成了。

2-1

在1-3的t++的每一时刻实时演算的模型下，2-1加入时间变量也比较好处理了，只需要在每一次判断是否需要接送人时增加判断该乘客的时间变量是否比目前的时间小即可。但是还有一个问题，即电梯不是全知视角，他如何区分乘客的楼层数据，是已经按过电梯的乘客的还是没有按过的。实际上这个问题在之前就遇到过，乘客的目标楼层是只有进入电梯才会输入的，并不能一次性全部读入，故而我新引入一组数据，在每一次有人被接入，则将这个人的目标楼层读入对应的第三个数组，这样每次检查是否有要送的乘客检查第三个数组即可。

时间也是如此，每当时间累加到与乘客的时间变量一致时，将此乘客的所在楼层读入一个新的数组，检查是否接人时检查此数组即可。

2-2

由于时间原因，此题最终并未完成，但大体思路已经成型。将电梯分为主副两台，1主2副，主电梯先按照2-1的程序运行，若出现：随着时间推移新读入的数据在电梯运行方向的后方（例如电梯处在上行模式4楼，此时2楼读入了一个呼叫请求）或乘客的去向与电梯的运行方向相反，则将该乘客的信息（所在和目标楼层）读入两组新的数组（副电梯读取的数组），并破坏掉他在原来数组中的数据（使主电梯再一次扫描时不会再考虑他的数据），并使n--，N++（n为乘客总人数，N为副电梯要运输的总人数）当n==0时，主电梯完成任务进入待机状态，副电梯开始运行（实际上是开始计算并输出关键信息，他是在接收到第一组被主电梯拒收乘客的数据后就开始运行的），他负责接收那些被主电梯拒收的乘客，他们的数据全在两个新数组中，让副电梯完全按照2-1的程序运行，只需加入在接人时若乘客去向与运行方向相反则continue此人（不抹去），直至N==0，运行结束。

但是此思路问题颇多，并未使两部电梯充分协调，会在正常转向时把转向楼层对应的乘客拒收并移交给副电梯，导致转向异常。因为时间有限这些问题我还未解决。

总结

此次招新考核题目让我受益多多。第一次自学了一门全新的知识，在看书、网课输入与自己试写简单程序中逐渐加深对知识的理解，最终用最基本的语句写出了比较合理的电梯程序，这个过程让我对c语言有了一定程度的掌握，并增强了我的信心。但是问题也很多，代码臃肿啰嗦，为了修复bug而各种修补导致可读性不强。

红色为关键思路，下划线为遇到的问题