* **开始前的准备工作：**

我虽然是0基础的大一新生，但因为以前玩过MC命令方块，做过PVP小游戏，对编程的思路多少有些了解。初看题目，我想这不就是个不断找最近的目的地的程序么，于是跃跃欲试。然而从学长学姐的口中了解到，学C语言要么得看动不动十几个小时的视频，要么得啃书，书里还讲得特别抽象。显然，dian的deadline让我不可能先系统的学完C语言再做。万幸dian在题目里给了菜鸟教程这个面向小白的网站，使我迅速掌握了一些基本语法，再加上稍稍浏览一下书本，大致思路就已经形成了。至于其他细节问题，我打算边学边做。个人认为只有不断地试错，而不是读死书，才是学习编程语言的高效途径。还有，学习一样全新的东西最好的流程或许是先全部浏览一遍这个领域的内容，学会抓取重点，先学透核心部分，然后其他知识就能融会贯通了。基于这个思路，我打算先跳过git、编译和文件读写的部分，从1-1开始，先尽快把程序的主要功能实现。

* **1-1：**

这道题目非常简单，scanf一下printf一下就ok。

* **1-3：**

从本题开始上难度。乘客数量一多，就要解决电梯下一步去哪的问题。一开始我没读调度算法那篇文章，还是想用排序算法不断找最近，自然就想到用一个数组表示电梯运行的目的地，然后用排序算法在这个数组中寻找最近的楼层。同时为了实现在电梯接上乘客以后再告知电梯这位乘客的目的地，我想到一个巧妙的办法。定义两个数组，一个记录乘客的起始楼层，一个记录目的楼层。起始楼层数组同时是电梯的目的地数组。每当电梯接上一位乘客，就把这位乘客对应的起始楼层数组中的数据替换成目的楼层数组中对应的数据，同时将目的楼层中对应的数据替换成一个标记数如100，相当于Null。这样就实现了在电梯的目的地中删掉刚刚接客的楼层，同时添加这位乘客的目的楼层。这样做还有个好处，就是乘客的所有状态，包括等待中，在电梯上，已到达目的楼层三种状态，都可以在这两个数组中体现出来。这对做超载和后续的两部电梯的分配问题很有帮助；还有就是，这样可以支持同一楼层有多个乘客上下电梯的情况，因为每个乘客都对应数组中的两个数据，而不是只记录了该乘客所在的这一层楼。这是以乘客信息作为电梯目的地的载体，而不是直接以楼层为目的地，自然就支持多个乘客上下了。

在屎山写成之后，我才发现有篇调度算法的文章没看。文章里清清楚楚的点明我这种不断找最近的算法的缺点，也就是乘客等待时间的方差特别大。结合常识，我发现好像确实没有电梯是不断折返找最近的。于是只好回去修改屎山，改成更为合理的LOOK算法。因为两个数组的机制可以很好表示乘客状态，我还是保留了下来。由于不想做太大改动，于是定义了电梯运行的方向，目的地数组中只保留了同方向楼层的数据，这样就基本符合了LOOK算法。此处省略后来痛苦的debug过程…（debug的时间有写代码的三倍以上，有时实在找不出就只能在每个关键节点print个记号出来，折腾半天最后发现竟是if里少了个等号之类的…）

再提一下超载的机制。当电梯人数达到4个以后，将所有未接的乘客对应的数组数据改成100，相当于没有数据，这样电梯就能直接忽略这些楼层，直奔着下人的楼层去。人数变回3个以后，就从备份数组里回复原来待接乘客的数据。这个备份数组是电梯的每次运行单位都会刷新的。

至此1-3完成

* **2-1：**

本题和1-3相比只是多了个用户输入的时间，但为此我又要经历一次大改。原来我也不是没有看到后面Level的要求，只是觉得时间好解决，只要计算楼层间的差值就好了。确实如此，虽然有一种特殊情况，就是电梯静止的时候也要计算时间，但也很好解决，只要单独针对这种情况计算时间即可。但致命的是，当我开始想2-2怎么做的时候（一开始我都没想到我会做出这么多，也就没怎么考虑2-2），发现如果不模仿真实世界，以时间推进为电梯的运行单位的话，两部电梯就没有统一的时间，也就不能实时分配目的地楼层给两部电梯。

为了适应2-2的要求，我只好将未完成的2-1大改，以for循环一次为一秒，电梯在这一秒内上楼或下楼，当检测到该层楼是目的地时就上下人并且显示信息。这涉及到很多改动和随之而来的bug，但好在我都顺利解决了。最终做完的2-1还是挺完整的，并且已经为2-2的编写做好了准备。

* **2-2：**

两部电梯的运行，最关键是解决如何把每位乘客分配给合适的电梯负责的问题。分配的依据显然是两部电梯到达每位乘客所在楼层所用的时间，哪个时间短就分配给哪部电梯。而时间则直接用电梯当前所在楼层和目的楼层计算出来（当然要考虑方向，如果有折返就加上折返所用的时间）。为了方便计算这个时间，每次电梯上下一层楼，就会把目前的所有待接乘客信息复刻到一个总的乘客信息数组中，用for循环计算电梯到达每位乘客所需的时间，哪部电梯用的时间少就把这位乘客的信息重新录入这部电梯的目的楼层数组中。这样就实现了实时分配乘客信息，两部电梯就能互不影响高效运行了。

至此2-2完成。

* **学习git：**

就和语法一样，我git也只学了点皮毛，目前唯一的应用就是回退版本，防止程序写得一团糟时回不去。根据网上的介绍，git在开源项目中起很大作用，大大方便了多人合作写代码。由于没有相关经历，git更高级的用法暂时还没有学习到。目前只会用git pull; git push; git add . 和git commit -m “”以及用git GUI回退版本。

* **学习编译过程 ：**

一开始就用了vs，后面也没啥时间了，没学gcc，只是知道有编译这个过程，把C语言的代码转译成机器语言，生成exe。后来发现vscode也可以用vs内置的编译器，于是就成功把.c文件编译成了exe。然而每次编译都要从命令行code .启动vscode，暂时还不知道怎么配置才能省去这一步

* **一些反思：**

我这个程序写了400多行，但经过后续学习，发现还有很多优化空间。比如可以用多维数组，记录乘客信息更加方便。还有可以调用函数，这样就避免重复写一些简单的部分比如排序算法。还有指针也没用上，到了deadline也还没有学明白。结构体也没用上，使得变量过于杂乱，不方便别人理解。毕竟我偷懒了，只学了菜鸟教程上的一点点语法，暴力地把程序写出来。以后还要系统的学习C语言，才能写出更高级的东西。总之，路漫漫其修远兮吧。