Elevator Project Report

Instruction

To use this simulator you can just paste the code into Visual Studio if you want to change some parameters, or just run the exe file.

You should only modify these parameters, or there may be some untested bugs.

capacity: most passengers a cart can hold

generatePassenger: time between generating passengers

tOpenDoor: time for door to open and close

tMovePassenger: time for a passenger getting on or off the cart

tMoveCart: time for cart to move one floor

tMaxWait: max time for cart to wait at the first floor

maxCome: max people come to the building when generating passengers

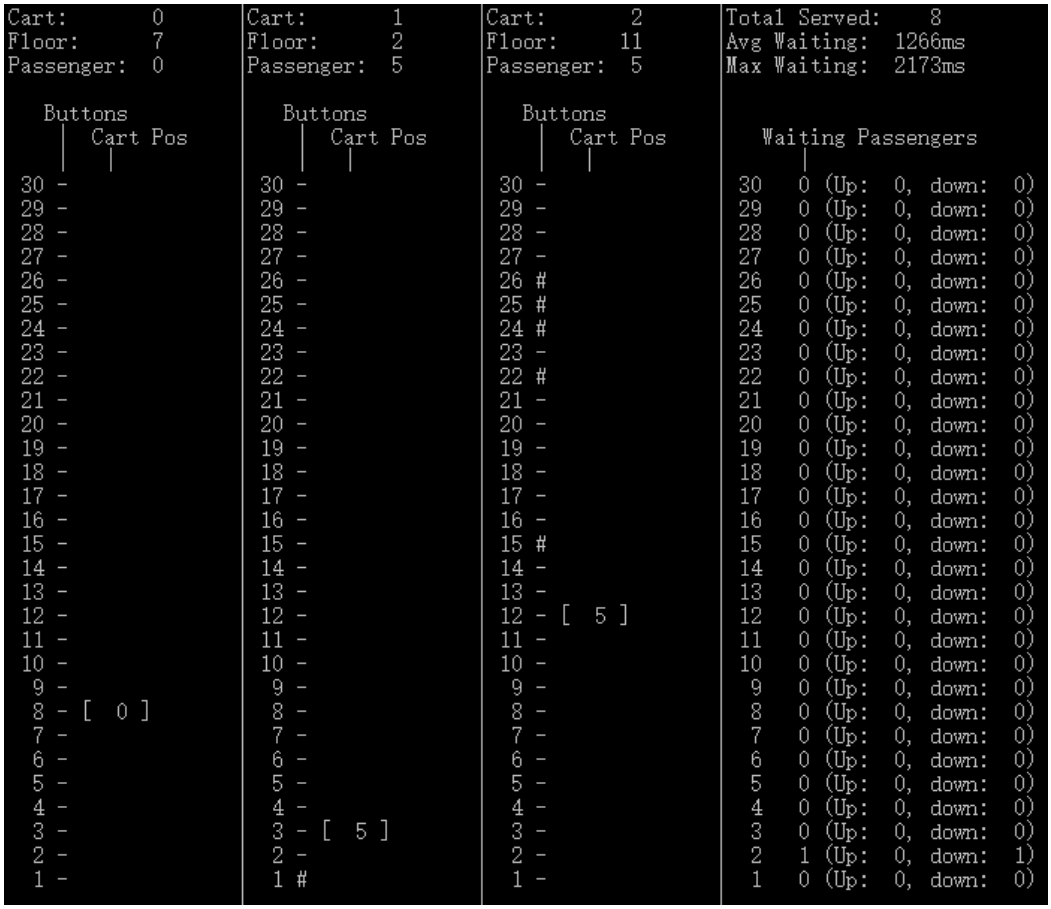
maxMove: max people want to move when generating passengers

tRunning: the simulation running time

stayAtFirst: whether the cart should go back to first floor while waiting

rFirst: the rate of going to first floor

The running result of this Program should be like this.



The first three columns are the visualized carts. In these columns, there are three smaller columns stand for floor number, the button in the cart and the position of the cart. For the buttons part, '-' stands for 'not pushed' and '#' stands for 'pushed'.
The last column is for the data analysis. The number of waiting passengers is shown in this section and we can also how many of them are going up or down.

Problem Description

In this project, there are three carts and 30 floors. It's easy to change the numbers, however the visualization part is not ready to adapt to these changes. The cart moving speed, door opening/closing speed and passenger moving speed can all be changed. Changing these parameters, we can simulate the operation of the elevator at different times.

Basically, when a cart get to one floor it will check if it should stop, which is checking if there are passenger waiting outside and their direction is same as the moving direction and checking if the button of this floor is pushed. When a cart stop, it first let the passenger, whose destination is its current floor, get of the cart, and then let the waiting passengers get on. Then, it can go to its next floor and do the same checking.

The capacity of the cart can also be changed. I set it to 15, which is the capacity of the elevator in my apartment.

To analyze the performance of the elevator, I choose three parameters. They are number of total served passengers, the max waiting time and average waiting time of the passengers. I think maybe the average time for each passenger to move one floor is also a good reference, but I haven't implemented that yet.

The logic of the cart running is simple, and there is only one logic for each cart. The three elevators are equivalent. The elevator will sleep until someone push the button. When a cart is full, it will wake another cart up. The elevator will first go to the top floor. When there are no passenger waiting at the higher floors and the cart is empty, it will turn back to the lowest floor and do the same checking. If we don't want the cart get back to the first floor it will fall asleep. Or, it will get to the first floor then sleep.

The visualization part is implemented with windows console. We can see the changing of number of passenger on the cart and waiting and also the cart moving.

Threads

There are for thread in my implementation, three threads for three carts and one for the elevator control.

The thread for control is also the main thread, this thread will initialized the environment including the console (visualization) initialization, which is set the window size and print some basic information and Dividing line. It then creates three threads for carts. After that, it will start generate passengers. For each period of time it will generate two types of passengers, passengers from first floor to other floors, which means coming to the building, and from other floors. There is a rate for the passenger created at the floors except the first floor to go to first floor, which means leaving the building. Each time passengers are generated, one

thread will be notified. This thread will also check if the program should stop. The analysis data is updated in this thread.

The cart thread is basically simulating the operation of cart. The logic of cart moving is described in Problem Description. Each cart is also responsible for modify the console buffer to show its changes.

Data Analysis

Here shows the data I run with different parameters in 60 seconds. The bigger maxCome and maxMove are the busier this time period is.

stayAtFirst	false		true	
maxCome/maxMove	8/16	20/40	8/16	20/40
Total Served	464	932	461	922
Avg Waiting Time	3791ms	5380ms	2748ms	4969ms
Max Waiting Time	8586ms	19177ms	6565ms	19043ms

We can see that the elevator performs well when there are few people. When we make the cart always stay at first floor the waiting time will be less. When it's busy, the elevator will always go to the first floor so stayAtFirst doesn't make much change.

Optimization

In my implementation, the three carts are the same. Perhaps, with a high building, different cart should take responsibility for different floors. When there are too many passengers waiting for elevator, the three carts almost move together. I haven't figured if that is a good thing but I think it would be better if the three carts can move separately, so the max waiting time can be lower.