OOP - Assignment 1

Eitan Kats 313356783
Ori Howard 208201939

Studying the problem **3**
Offline algorithm: 4
UML: 4
Code Flow: 5
Initialization: 5
Allocation: 5
4. Final results **6**

1. Studying the problem

First article:
Cities 101: How Do "Smart Elevators" Work?
“The passengers first enter their desired floor as they approach the elevators. The keypad sorts them into groups of similar destinations and assigns specific elevators to each passenger. So passengers going to floors 26, 28 and 32 would be assigned one elevator, while passengers who keyed in floors 50, 54 and 55 would take another. In cases when every second is necessary, the Schindler elevators can also detect employees via their ID badges.”

“On the way down, full elevators skip floors to minimize wasted stops. The Miconic monitors each car’s current weight, and ceases to make stops once the weight passes a certain limit. The system also tracks traffic patterns and remembers the most frequently called floors. The decrease in trips preserves equipment longer as well”
Second article:
Everyday Algorithms: Elevator Allocation

Third article:
The standard elevator algorithm follows a protocol defined by adherence to three basic rules: (1) the elevator continues in its current direction as long as there is demand in that direction; (2) the elevator switches direction whenever there is no more demand in the current direction and there is demand in the opposite direction; (3) it stops and waits for the next call otherwise (information taken from here)

Fourth article:

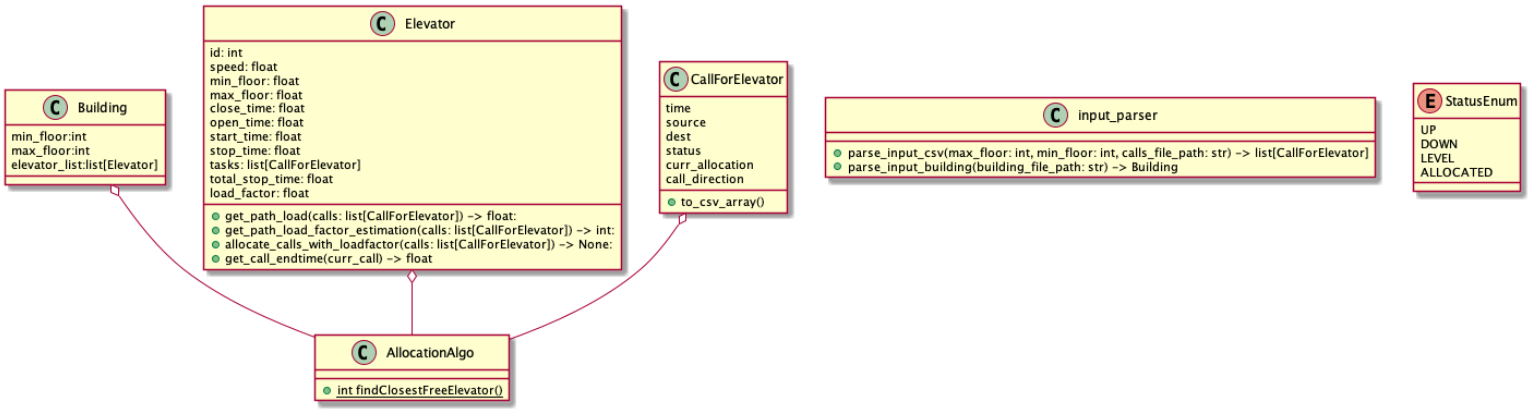
https://softwareengineering.stackexchange.com/a/331724/374093

Note that once you get to a building with 100 floors, you will typically have elevators serving only a specific range of floors (e.g. 0-19, 20-39, …), as well as express elevators that only go long distances (e.g. 0 to 50, 0 to 100, 50 to 100, but no floors in between), so you might have to change elevators to get to your destination. You might also have multiple elevators per shaft that obviously cannot pass each other. Totally off-topic: IIRC, there was a question about the efficiency of those up and down arrow buttons on the User Experience site that made for very fascinating reading.

Offline algorithm:

Because we know in advance our input we perform a calculation on the data to be more efficient.

UML:



In this section we will describe the way our offline algorithm works:

First of all we define the variable that we use in the algorithm:

1. Total_stop_time - this is the time is takes for an elevator to stop at a given location this is a sum of the following parameters:
    a. Close_time
    b. Open_time
    c. Start_time
    d. stop_time
2. loadFactor - the load factor is holding the load of each elevator which is calculated using the calls that are currently in the elevators' queue with regards to the location of the elevator.
    The factor is calculated using the following formula: $\frac{distance}{velocity} + total\ stop\ time$ for each call
    The distance is the sum of the distance of each call with regards to the location of the elevator

Code Flow:

Initialization:

The initialization process works as follows:

1. Attempt to parse command line arguments
2. Use the command line arguments to parse the files given as input
    a. Call files that don't match a building will output a message saying that they don't match
    b. Files that don't exist will also trigger a message saying that
    c. Any other error will ask you to check your input for the script
3. If the input was a valid input we parse it into objects (building and a list of calls)

Allocation:

The allocation process is the following:
We are iterating over the parsed calls and each call that is not allocated is going through a pipeline which tries to find future calls that are going to be allocated during the movement of the elevator, this path building process is executed on each elevator.
This is the part where we utilize the fact that we have the information beforehand
Each path is then filtered using a function that simulates the movement of the elevator according to the elevators' parameters.
Then we check which elevator can complete its path in the fastest time according to the load_factor(defined above) of the elevator.
The chosen elevator then executes the path that was built for it

4. Final results

Calls_b	B5	Total waiting time: 41196.0	average waiting time per call: 41.196	unCompleted calls:0	certificate: -255151820
Calls_c	B5	Total waiting time: 41674.0	average waiting time per call: 41.674	unCompleted calls:0	certificate: -255151820
Calls_a	B5	Total waiting time: 1730.0	average waiting time per call: 17.3	unCompleted calls:0	certificate: -70972548
Calls_d	B5	Total waiting time: 41518.0	average waiting time per call: 41.518	unCompleted calls:0	certificate: -255151820
Calls_b	B4	Total waiting time: 181706.49462399984	average waiting time per call: 181.70649462399984	unCompleted calls:13	certificate: -570735033
Calls_c	B4	Total waiting time: 184201.76122	average waiting time per call: 184.20176121999998	unCompleted calls:4	certificate: -580558754
Calls_a	B4	Total waiting time: 1997.0	average waiting time per call: 19.97	unCompleted calls:0	certificate: -78386090
Calls_d	B4	Total waiting time: 195238.874732	average waiting time per call: 195.238874732	unCompleted calls:2	certificate: -795245746
Calls_b	B3	Total waiting time: 530281.8701440002	average waiting time per call: 530.2818701440002	unCompleted calls:128	certificate: -1784768488
Calls_c	B3	Total waiting time: 537092.9335500026	average waiting time per call: 537.0929335500026	unCompleted calls:110	certificate: -1777155078
Calls_a	B3	Total waiting time: 3071.0	average waiting time per call: 30.71	unCompleted calls:0	certificate: -23720315
Calls_d	B3	Total waiting time: 538773.2333840034	average waiting time per call: 538.7732333840034	unCompleted calls:124	certificate: -1775701028
Calls_a	B2	Total waiting time: 5028.0	average waiting time per call: 50.28	unCompleted calls:0	certificate: -219123959
Calls_a	B1	Total waiting time: 11292.0	average waiting time per call: 112.92	unCompleted calls:0	certificate: -273190417