Assignment Zero – OOP :

Alon Barak – 213487598

Omer Adar – 325022952

Literature review:

Projects we thought might be helpful for this assignment:

1. Efficient Elevator Algorithm - Tennessee University. <https://trace.tennessee.edu/cgi/viewcontent.cgi?article=3380&context=utk_chanhonoproj>
2. Smart Elevator – Geeks For Geeks.

<https://www.geeksforgeeks.org/smart-elevator-pro-geek-cup/>

1. Intellevator – IEEE Access.

<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9003300>

The difference between Online algorithms and Offline algorithms:

Most of the time, offline algorithms send the entire data as an input to the algorithm and the method can use the data to optimize and solve the problem better.

On the other hand, online algorithms sends the data piece by piece in a serial order without having the entire data from the beginning.

As a result of not having the whole data from the beginning, online algorithms might make decisions that later on might seem not so smart or optimal. Offline algorithms have the option to make the best decisions according to the given data which make it easier for offline algorithms to reach a better and more efficient solution than online algorithms.

Off-line algorithm for the elevator optimization problem:

Assuming we receive an array of N calls for elevators as an input to the algorithm, U calls up and D calls down (U + D = N).And assuming we have a number of K elevators. Each call have source floor and destination floor.

First, lets divide the array into 2 arrays of UP calls and DOWN calls, lets say u is the length of the UP array and d is the length of DOWN array. We will allocate the elevators for each direction by the ratio of u:d. in case there is only one elevator it will take care of all the calls (up & down). After the division into directions the calls will be divide between the elevators based on the speed of each elevator individually – fast elevators will receive more distant calls than the slower elevator.

On-line algorithm for the elevator optimization problem:

The algorithm will base on the nearest elevator with capacity consideration to the call's floor. Assume we have N calls and K elevators, the Capacity of each elevator will be For each call we will find the nearest and fastest elevator, then we will have next options: