Algorithms and Data Structures 1

Student names: Jacopo Ceccuti s215158
Hand-in: Floor Plan

1 Exercise

1.1

In order to model the floor plan as a graph we can simply set that every room is represented by a vertex and every door by a node.

1.2

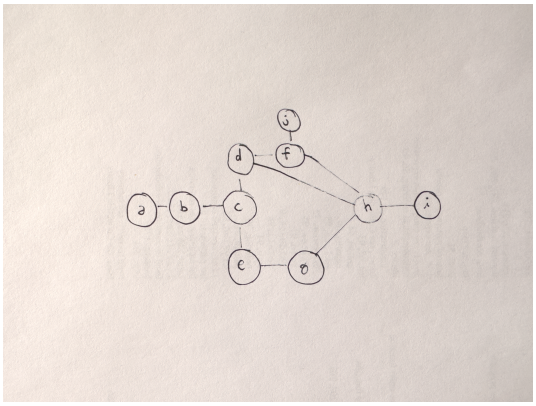

Figure 1: Floor plan P

1.3

To solve this problem we firstly need to identify which doors are "normal" (i. e. not fire doors). To do so we can get the set difference between the set D of doors and the set B of k fire doors (this takes $O(D)$ time). Now we can use a dynamic connectivity, with union find, data structure and do R "INIT" and $D - B$ "INSERT". Then we can use DFS or BFS to run through the graph starting from the entrance and check if the numbers of vertices visited at the end is equal to R, in that case it means that all of the rooms are connected and the evacuation is still possible. The running time is therefore:

$$O(D + R + log(D - B) + (D - B + R)) = O(D - B + R)$$

1.4

This last task asks us to change our model in a undirected weighted graph that can represent the cost of the floor plan. Assuming that the graph is already created we can simply run Kruskal's or Prim's algorithm. Taking for example Kruskal we will have the following running time for: sorting D edges, 1 "INIT", D "CONNECTED" and R "INSERT".

$$O(D \cdot log(D) + R + D \cdot log(R) + R \cdot log(R)) = O(D \cdot log(R))$$