Assignment 2 Report

Description

For this Project we had to make a program able to apply the A* algorithm to find the closest path from an starting point to a goal point.

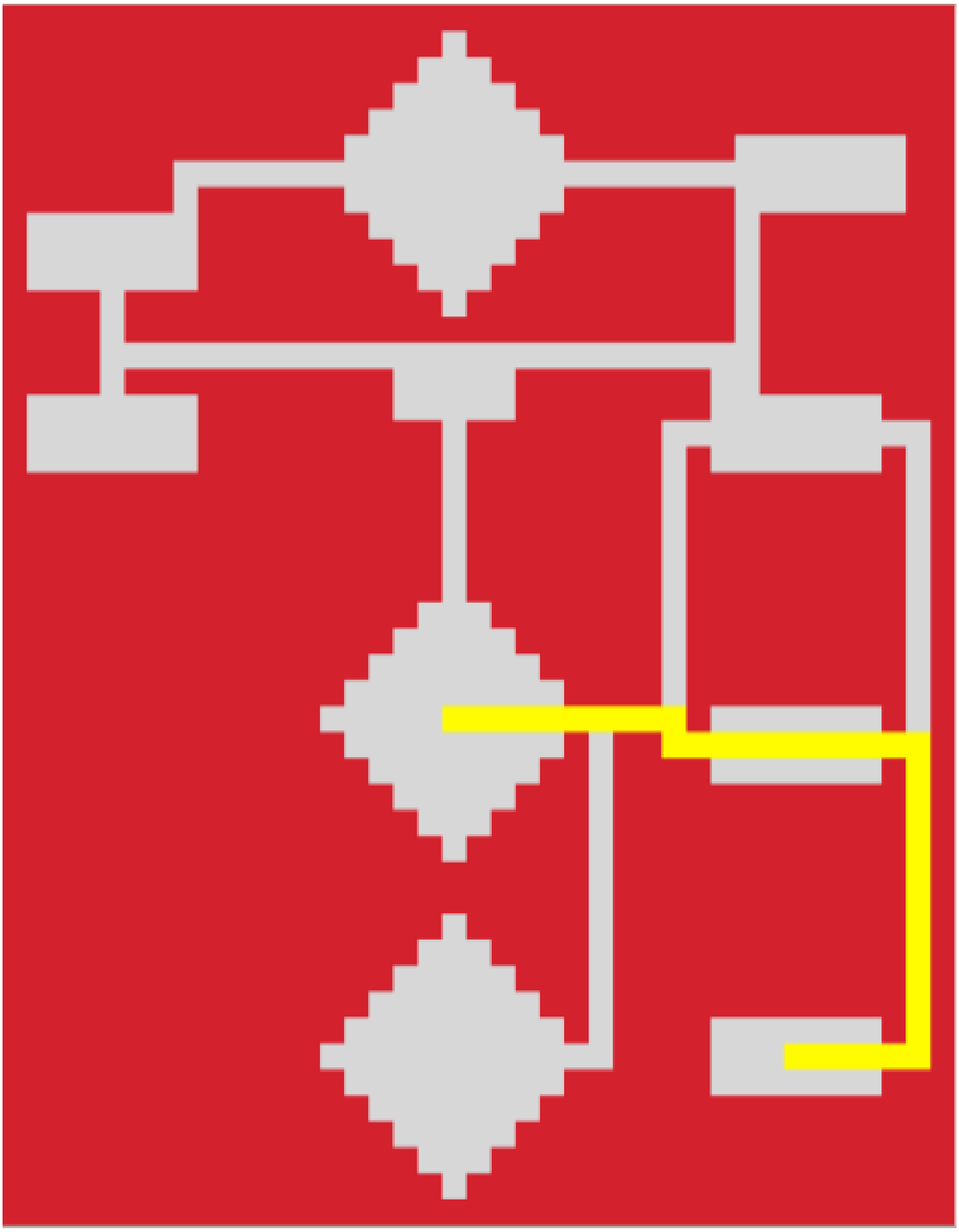
For that, first we define a class node that includes a position, the cost to get to it, wich is divided into the cost of getting to the point, the heuclidean distance and the total cost; and the previous point. Then we made the functions to calculate the euclidean distance and another to calculate which is the best node to examine next. After that we get to the A* search in which we follow the following steps:

- First, we make three arrays storing nodes, one for the resulting path, one for the nodes to be examined, and another for visited nodes.
- Second, we set up the start position with its total cost and including the not_visited and the visited array, as well as setting it as the current node to examine.
- Then the loop starts by examining the neighbours of the current node, calculating there costs and adding them to a neighbours list if they are not a wall.
- After visiting the 4 nodes we check if they were already in the visited or not_visited list and they weren't, we add them in the not_visited.
- Finally, we set a new current node by using the findBestNode function and we check if it is the goal. If it is, we reconstruct the path checking the previous node of each node leading up to the goal until we reach the start; If it isn't, we examine the new node.

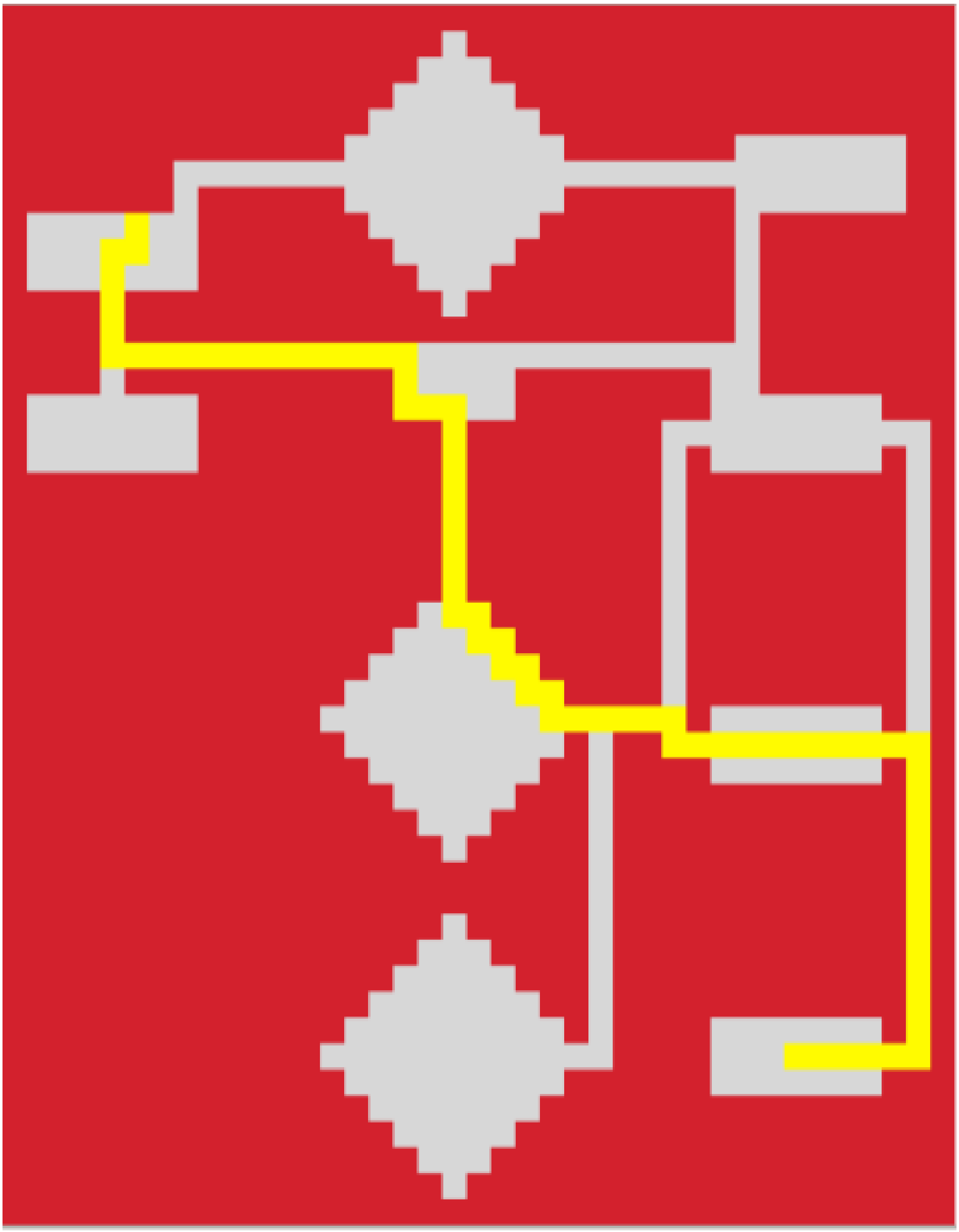
After finding the best path, we use a modified version of the function provided by the assignment to include a yellow colour to show the path.

Results

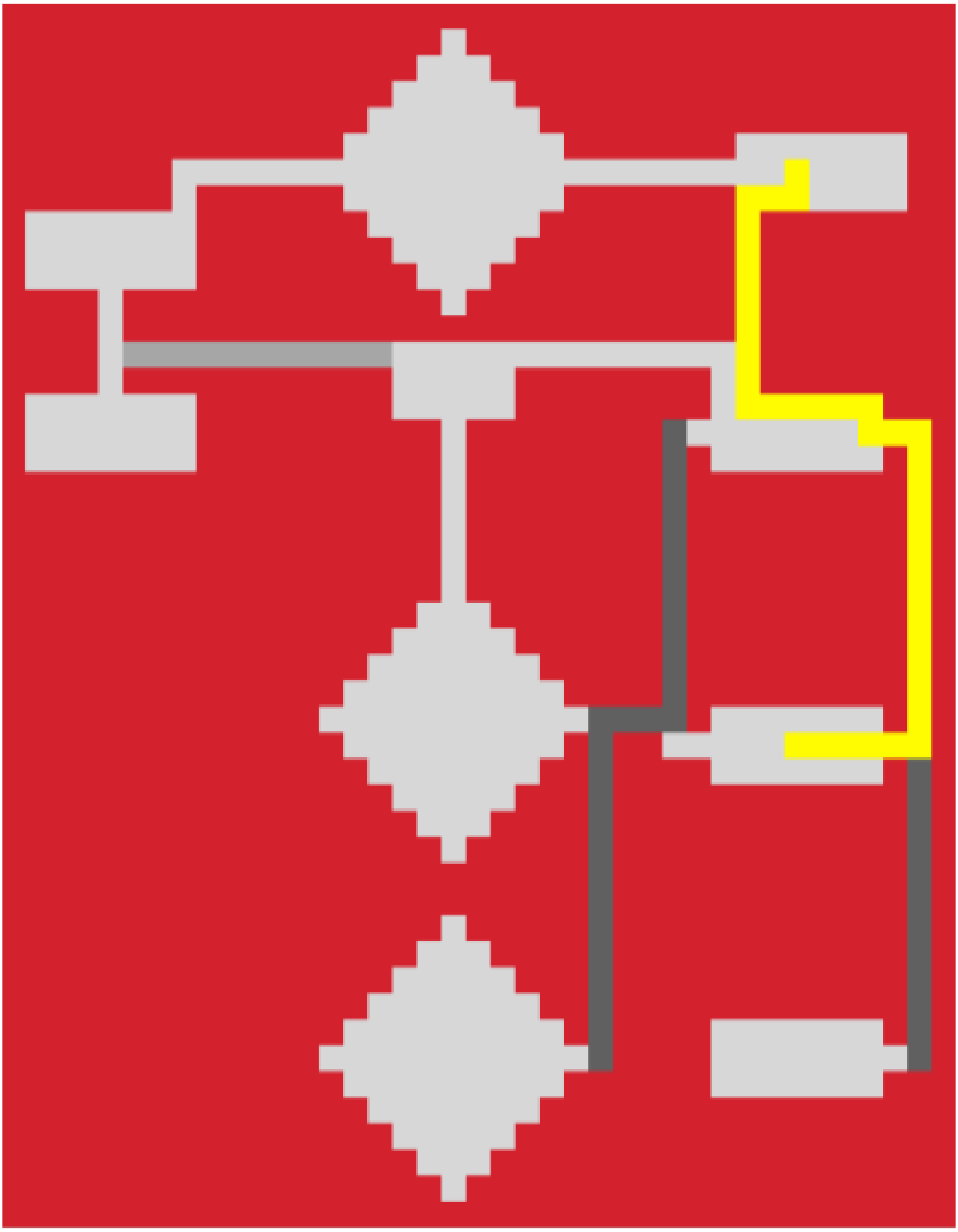
Task 1



Task 2



Task 3



Task 4