COMPUTER ENGINEERING DEPARTMENT

COMPUTER ENGINEERING

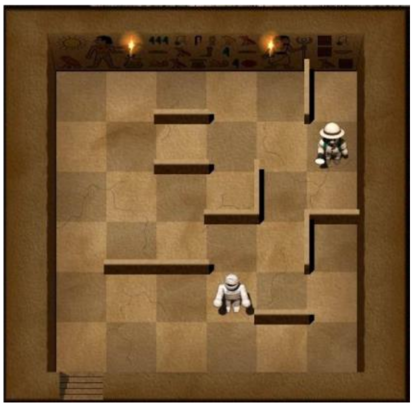
ARTIFICIAL INTELLIGENCE

2021/2022 – 2º semestre

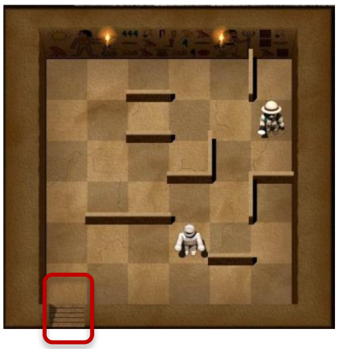
PROJECT – MUMMY MAZE SOLVER

1. Mummy Maze

Mummy Maze is a game where the hero is a treasure hunter. While searching for lost treasures, the game hero goes through several levels where he must avoid being caught by enemies and traps.



In each level or chamber there is a passage to the next level/chamber. The goal of the hero is to reach that passage avoiding threats: being caught by and enemy or falling in a trap. The passage to the next level is the cell next to the stairs (see next figure).



This is a turn-based game, where, in each turn, the hero moves first, and only after that the enemies move (even if the hero has reached the goal cell).

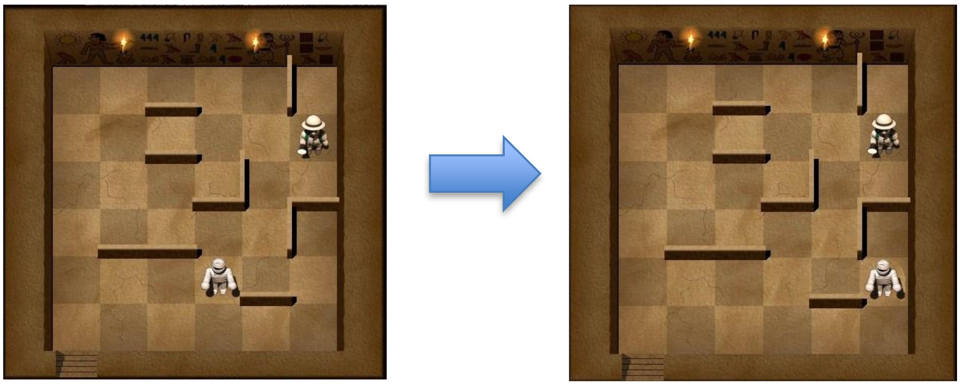
2. Game characters and elements

Hero

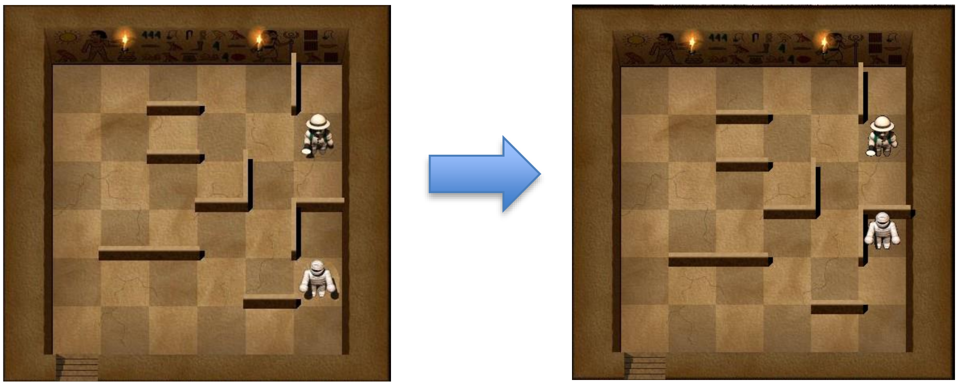
The hero is able to move just one cell each turn, up, down, left, right, unless there is a wall blocking the intended direction. The hero can also opt for not moving in its turn.

White Mummy

The white mummy moves up to two cell in each turn. The goal of the mummy is to catch the hero, and, for that, it will try to move to the cell where the hero is. The white mummy tries first to move to the same column of the hero and only after that it tries do move to the same line where the hero is. Using the above example, if the hero doesn't move, the mummy would move two cells to the right (see next cell).

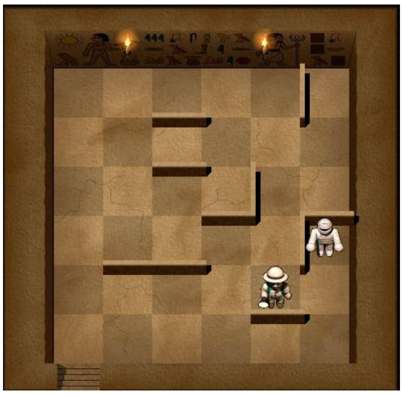


If the hero stays still in the next round, the mummy will be as follows:



In this case, the White mummy move just one cell: as it is already in the same column of the hero, it tries to move to the same line going up; after moving one cell up, the white mummy finds a wall that prevents it from going further up.

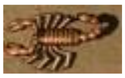
In the situation depicted bellow, the white mummy tries to first move to the right, but it is not able to do so. Since it has still two more moves, it moves down one cell. Since it still has one movement, it moves one cell to the left and kills the hero. That is, the mummy always tries to spend its two movements, giving priority to the columns.



Red Mummy

Such as the white mummy, the red mummy can do up to two movements each turn. The white mummy tries first to be on the same line as the hero and only then in the same column.

Scorpion



The scorpion moves in the same way as the white mummy but it only moves at most one cell per turn.

Key

When the hero passes through the cell containing the key, he opens or closes a door. If the door is open, it closes and vice-versa.

Door

When it is closed, the door behaves like a wall: it locks the hero movement, as well as the enemies movements.

Trap

The hero dies when it steps into a trap. Traps have no effect on enemies.

3. How to kill mummies

In some levels where there are two mummies, it may be necessary to kill one of them in order to succeed in finishing the level. If the hero is able to lead the mummies to occupy the same cell, they will fight to one another and only one of them survives. The dead mummy just disappears from the level.

4. Work to be done

You should develop an application that uses search algorithms to play the Mummy Maze. You should develop at least two distinct heuristics suitable to the problem to be used by informed algorithms. The application should allow the user to choose the level to be solved, the algorithm to be used and the heuristic (if applicable).

You should do a comparative study of the performance of the different algorithms and heuristics. Namely, you should compare:

- The performance of the different non-informed algorithms (with and without optimizations);
- The performance of the different informed algorithms (with and without optimizations);
- The performance of non-informed algorithms versus the performance of informed algorithms;
- The performance of the developed heuristics.

5. States representation

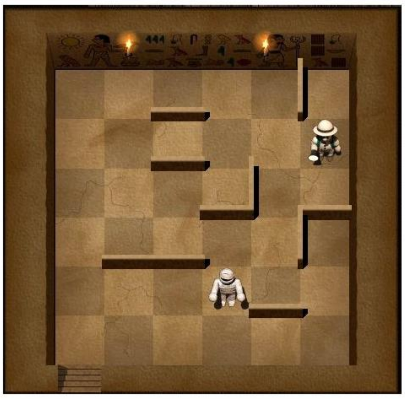
There are many Mummy Maze levels. In this project the program to be developed should be prepared to play only level with the dimensions of the level presented above in previous figures, with dimension 6 x 6. However, it should be able to define different level configurations. For that, text files should be used, which define the levels, namely, the hero and enemies initial position, as well as walls, keys, doors, traps and the goal position.

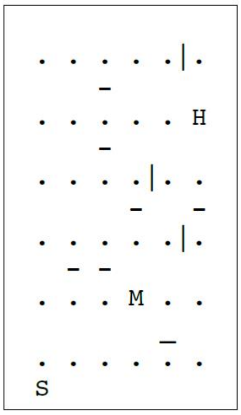
These files should define a 13 x 13 characters file where each character represents a level cell, a wall or the goal position. The characters to be used are as follows:

- '|' ou '–', to represent a wall;
- 'H', to represent the hero;
- '.', to represent an empty cell;

- 'S', to represent the goal position;
- 'M', to represent the white mummy;
- 'V', to represent the red mummy;
- 'A', to represent a trap;
- 'E', to represent the scorpion;
- 'C', to represent a key;
- '=', to represent a closed horizontal door;
- '_', to represent an open horizontal door;
- '"', to represent a closed vertical door;
- ")", to represent an open vertical door.

For example, the level below is represented as follows:





```
 . . . . .|.
     -
 . . . . . H
     -
 . . . .|. .
       -   -
 . . . . .|.
   - -
 . . . M . .
         -
 . . . . . .
 S
```

In the course website, in the same directory of the project sheet, there is a file named `niveis.zip` with some levels in this format to be solved. Students are encouraged to define new levels and even publish them in the course forum so that they can compare the results obtained with different solution and heuristics to the same problem.

6. Solutions visualization

The `MummyMazeShowSolution.jar`. library is provided in order to allow a nicer visualization of the solutions. This library should be included in your application. It contains the `SolutionPanel` class, which belongs to the `showSolution` package. You should call the `showSolution(List<String> states, double solutionCost)` method of this class. This method is responsible for showing graphically the sequence of states that result from the application of the found solution using the search algorithms, as well as the cost of the solution. As first argument of the method, you should pass a list of strings corresponding to the sequence of states that result from the application of the solution. The second argument should be the cost of the solution. The strings must be according to the format described above.

7. Report

The report to be written should include:

- A description of the representation of the states;
- A description of the developed heuristics;
- A presentation and discussion of the obtained results;
- A description of the contribution of each member of the group in the development of the project and in the report writing;
- Other aspects considered relevant for a good understanding and assessment of the developed work.

Some of the most important factors in the report assessment include:

- Clarity in the description of the application components;
- The way how the results are presented (using tables and/or charts may help);
- The analysis and discussion of the results.

8. Assessment

10% - Problem definition
20% - State definition
15% - Actions definition
20% - Heuristics
20% - Comparative study and report
15% - Extras

Suggested extras:

- Implementation of specific optimizations to some of the algorithms developed in the classes in order to turn them more efficient (students should be able to identify what algorithms can be optimized as well as how can they be optimized);
- Implementation of other search algorithms;
- Running sets of experiments automatically;
- Implementation of more heuristics.

9. Deadlines, dates, rules and instructions

1. Project delivery deadline: **June 21st 2022, 23:59**.
2. Oral exam date: **July 5th 2022**.
3. The project should be developed in groups of 2 students. Groups with more than 2 students are not allowed. Students that want to develop the project alone should as kit through email to the theoretical classes teacher. Only in well fundamented situations this will be allowed.
4. The report should be written using the template provided in the course Moodle page.

5. The project should be delivered as a zip, rar or 7z file containing all the project elements, including the report. The file name should follow the format IA_Projeto_#1_#2.(zip/rar/7z), where #1 and #2 should be replaced by the student numbers of the group elements. The report should be in the pdf format and its name should follow the same format (with pdf extension).