Department of Information & Computer Engineering

2022-2023 **5th Semester**

1st paper for the Artificial Intelligence lab

Application of search algorithms to the pacman problem

In the simplified world of the problem there are four consecutive neighboring cells, numbered from 1 to 4. In some of the cells there may be a unique fruit, but this is not necessary. Also, in any of these four cells there is a pacman that can move to an adjacent cell, left or right.

The permissible states in which the problem can be found are determined by the combinations of relations between its objects. A state is determined both by the pacman's cell in any of the cells, and by which cells contain fruit (see figure 1). Any admissible state in which the problem world can be found at the beginning of solving the problem can be defined as the initial state.

The goal of the problem is to eat all the fruits and leave pacman alone in one of the 4 cells.

Figure 1: *Various versions of the pacman world*

Job Requirements (in steps)

A) **Modify** the problem both in terms of its world and its functions, so that you make it more interesting. Make the grid of cells 2D by creating operators to move up and down. Also, optionally instead of one fruit (f) you can declare a number of fruits present in a cell, with a suitable modification of the fruit-eating operator, etc.

B) Solve the modified pacman problem using the **algorithm of depth first search (DFS) with frontal tracking** to find the target. Present and comment on the results.

C) Add parallel **queue monitoring** capability **of the paths** that DFS creates during resolution.

D) Solve the modified pacman problem by implementing the **breadth-first search (BFS) algorithm by following the queue of paths** to find the goal.
You can add the option to choose between DFS and BFS at the start of the search.

E) **Document your work appropriately** and submit it together with the code that will result at the end of all the above steps along with the **correctness checks** that you will perform separately at each step.

Guidelines for documenting work

1. Define and represent appropriately:
   - The world of the problem.
   - The initial and final state of the problem.
2. To describe the transition operators.
3. To determine the space of states of the problem (descriptively and with lists), citing representative examples.
4. To present an **annotated** coding (in python) of the problem world and the problem's transition operators, as well as the findchildren function.
5. Present the results from step D (implementation of breadth-first search (BFS) with path-queue monitoring) for 2 different initial states

The work is compulsory and individual. It will be delivered by 10/1/2023, 23:59 via e-learning platform eclass.uniwa.gr. Deliverable will be compressed file (zip or rar) with file name Surname-AM.zip (eg Tselenti-2434233.zip) which will include:

***(a)* the work documentation in pdf format.**

***(b)* the commented .py source code;**

The grade of this paper will contribute 20% to the final grade of the course, for them following the PADA program.

For those following a TEI program, the grade of this paper will contribute 50% to the final grade of the laboratory part of the course.

Explanation and analysis of the job requirements: during the theoretical course

Support for the implementation of the work: every Monday 16:00-18:00, room K10.022

Supporting videos for the implementation of the assignment can be found in eclass