HO CHI MINH NATIONAL UNIVERSITY

UNIVERSITY OF SCIENCE

---o0o---

**Introduction to Artificial Intelligent**

**A blue and white logo

Description automatically generated**

**Project 01**

**HIDE AND SEEK**

*Instructors:* Nguyễn Ngọc Thảo Hồ Thị Thanh Tuyến

Lê Ngọc Thành Nguyễn Trần Duy Minh

*Members:* Trần Anh Minh 22127275

Đoàn Đặng Phương Nam 22127280

Bùi Nguyễn Lan Vy 22127465

Diệp Gia Huy 22127475

**HO CHI MINH CITY, APRIL, 2024**

**CONTENTS**

# Assignment Planner

# Environment Requirement

*Python version:* 3.10+ with Pygame graphical module.  
Consider installing Pygame using *pip install pygame* if it is not available.

*Usage:* Run the application by executing *python main.py* in the console/terminal.  
*Notice:* The command may vary across platforms, the above command it tested on a Windows operating system.

# Idea and Theory

# Maps Design

To some extent, generating maps for testing the seeker has lots of things in common with letting the player find the way to solve a maze. Creating multiple paths around and placing obstacles at the correct position would require the seeker to run around the map several times.

# Problem

## Preparation

To prepare common attributes and methods which can be utilized in all levels of Hide and Seek game, we create a versatile class called **Level**, then in each level, we will simply create the corresponding class which inherits this class besides supplementing essential things for each specific level.

**Level** class efficiently initializes a general problem space with the map and score of game, the number of steps taken by the seeker and all of hiders as well as the current turn taker who can be the seeker or hiders. In addition, this class also defines the following methods:

broadcastAnnouncement: Receiving the hider of position which have the demand to broadcast an announcement in the map as the only additional input, this method returns a random position for the hider within radius 3, including potential obstacles or walls.

## Level 1

## Level 2

## Level 3

## Level 4

Though we did not implement level 4, we have some ideas about this.

One opinion is that, letting the hiders identify the corners, they should move the obstacles to fill up the entrance and hide inside. Therefore, it blocks the seeker from finding itself.

Another idea is to allow hiders to move the obstacles on their way, running away from the seeker if being chased, and block the path to have the seeker find another path to reach it.

However, blocking the path in the first idea may not seem like a valid choice for a non-reinforcement approach; the hiders need to study the map well enough to figure out where the dead end is located.

# User Interface and Game Play

A screenshot of a game

Description automatically generatedWhen running the program, a screen will pop up to allow the user to choose the desired level to run.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedAfter choosing a level to run, the screen changes, and a button is shown to let the user pick a map to run. When clicked, all available maps are provided for the user to choose and it will appear accordingly.

A screenshot of a computer game

Description automatically generatedA screenshot of a computer game

Description automatically generatedOnly then will a begin button pop up, allowing the user to run the game.

There can be two results: the seeker will find all hiders, or the seeker will give up (in level 3 or more, the hiders can make the seeker chased endlessly). And the begin button will be changed to "Restart", letting the user load the map again for another run.

# Evaluation

# Refs

<https://www.pygame.org/docs/ref/pygame.html>