**Artificial Intelligence**

**Project 1**

| Topic |

**Searching**

by

**PHẠM VŨ DUY - 18127092**

**MAI ĐĂNG KHÁNH – 18127118**

**NGUYỄN PHÚC THỊNH - 18127223**

**HUỲNH NHẬT NAM - 18127014**

1. **Overview:**
2. **Environment:**

* Python 3.8, with graphical library:

+ *pygame* library

* Divide the problem into small object:

+ Ghost

+ Pacman

* Using Object-Oriented-Programing

1. **The degree of completion:**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Specifications** | **Scores** | **Degree of completion** |
| **1** | Finish level 1 successfully. | **15%** | **100%** |
| **2** | Finish level 2 successfully. | **15%** | **100%** |
| **3** | Finish level 3 successfully. | **10%** | **100%** |
| **4** | Finish level 4 successfully. | **10%** | **100%** |
| **5** | Graphical demonstration of each step of the running process. You can demo in console screen or use any other graphical library. | **10%** | **100%** |
| **6** | Generate at least 5 maps with difference in number and structure of walls, monsters, and food | **10%** | **100%** |
| **7** | Report your algorithm, experiment with some reflection or comments | **30%** | **100%** |
| **Total** | | | **100%** |

1. **Assignment plant:**

|  |  |  |
| --- | --- | --- |
| **Student** | **Job** | **Scores** |
| Mai Đăng Khánh  18127118 | Graphical demonstration | 10% |
| Pacman in level 4 | 5% |
| Demo and testing | 15% |
| **Total** | **30%** |
| Huỳnh Nhật Nam  18127014 | Level 3 | 10% |
| Writing the report | 15% |
| **Total** | **25%** |
| Nguyễn Phúc Thịnh  18127223 | Ghost in level 4 | 5% |
| Create maps | 10% |
| Fix bug in level 2 | 5% |
| **Total** | **20%** |
| Phạm Vũ Duy  18127092 | Level 1 | 15% |
| Level 2 | 10% |
| **Total** | **25%** |

1. **Algorithm description:**
2. Level 1 and 2:

* Use *Breadth First Search* to find food:

+ *BFS* always find solution to a problem if it exist

+ BFS is very useful especially when finding the shortest and most optimal path to one edge

+ In level 1 and 2, the map is not close. With *BFS*, pacman can take advantage on this characteristic of the map and reduce the path to the minimum number of move

+ In level 1 and 2, pacman know the food location, so *A\** can also find the shortest and the most optimal path to food. But, without a admissible heuristic which can support not closed map, *A\** can’t do the search better than *BFS*

1. Level 3:
2. Level 4: