

18 - 2 인공지능 과제 1

IT/BT 탈출하기

조교 김병조

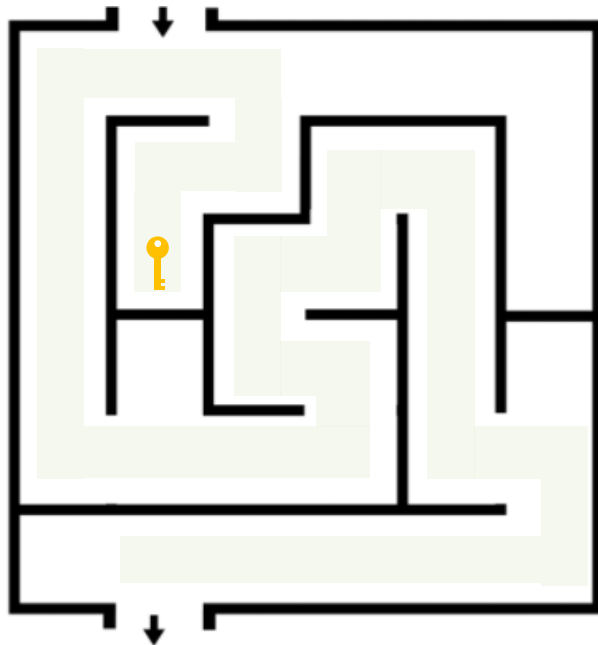
bjbj8083@gmail.com

과제 개요

- 다양한 Search Algorithm을 이용하여 여러 층으로 이루어진 미로 경로 탐색.
- 세부사항
 - 총 5개의 층을 탈출하는 것이 목표
 - 각 층에는 입구와 출구가 있으며, 통로와 벽이 구분되어 있음.
 - 현재 위치에서 상하좌우 방향으로 이동 가능
 - 출구는 자물쇠로 잠겨 있어 출구로 가는 도중에 키를 습득해야 함.
 - 알고리즘 : Breadth first search, Iterative deepening search,
Greedy best first search, A* algorithm 등
 - 각 층마다 사용할 경로 탐색 알고리즘을 선택할 수 있음.
 - 하나의 층에서는 한 가지의 알고리즘만을 사용.
 - 각 층의 최단 경로(length)와 탐색한 노드의 개수(time)로 점수 산출.

결과물 예시

- 출력 : 1,2,3,4,5 로 표현된 m*n 행렬 , 최단경로 길이, 탐색한 노드 개수
(1 = 벽, 2 = 통로, 3 = 출발점, 4 = 도착점, 5 = 최적 탐색 경로)



1	1	③	1	1	1	1	1	1	1	1	1
1	5	5	5	5	2	2	2	2	2	2	1
1	5	1	1	5	1	1	1	1	1	2	1
1	5	1	5	5	1	5	5	5	1	2	1
1	5	1	5	1	5	5	1	5	1	2	1
1	5	1	5	1	5	1	1	5	1	1	1
1	5	1	1	1	5	5	1	5	1	2	1
1	5	1	2	1	1	5	1	5	1	2	1
1	5	5	5	5	5	5	1	5	5	5	1
1	1	1	1	1	1	1	1	1	1	5	1
1	2	5	5	5	5	5	5	5	5	5	1
1	1	④	1	1	1	1	1	1	1	1	1

length=55
time=74

점수 산출 - 코드

- 코드 (70%)

- 파일 이름

- 본인학번_assignment_1.py (파이썬 파일 한 개)

- 구조

- 1층 – `def first_floor(...)`
 - 2층 – `def second_floor(...)`
 - 3층 – `def third_floor(...)`
 - 4층 – `def fourth_floor(...)`
 - 5층 – `def fifth_floor(...)`
 - 필요에 따라 작성한 기타 함수들

※ 함수들의 인자와 리턴 값은 자유 (단 보고서에 명시)

- Test

- 임의의 미로를 이용하여 실제 작동 여부를 평가하여 점수 산출

점수 산출 - 보고서

- **보고서 (30%)**
 - **코드 설명**
 - 각 층마다 **코드 실행방법**을 명시.
 - 각 층마다 **코드 동작**에 대해 설명.
 - **사용 알고리즘**
 - 각 층마다 **어떤 알고리즘**을 **왜 선택**하였는지에 대한 설명.
 - **최단경로(length)**
 - True / False (**최단경로가 맞는지** 확인하여 점수 산출)
 - **탐색한 노드의 개수(time)**
 - 미로와, 이동 방향의 우선순위에 따라 달라질 수 있음.
 - 오차를 계산하여 점수 산출

과제 조건

- 환경

- 프로그래밍 언어 : **Python 3.x**
- OS : **Ubuntu 16.04 LTS**
- 외부 라이브러리 사용 **불가**(내장함수만 사용 가능)

- 제출 사항

- 파이썬 파일 : 본인학번_assignment_1.py (파이썬 **파일 한 개**)
- 결과 보고서 : 본인학번_assignment_1.pdf
 - 코드 설명
 - 사용 알고리즘
 - 실험 결과
- 출력 파일 (5개) : *_**floor_output.txt** (*는 first, second, third, fourth, fifth)

주의 사항

- 파일명 반드시 준수.
- 파일은 **GitLab**에 올려주세요.
- 제출 기한 : **2018.10.1**
- 추가 제출 기한 **없음**.
- 점수 비중 : **코드 70% 보고서 30%**

주의 사항

- 아래 예시와 같은 입출력 형식을 반드시 준수.
- 코드와 같은 경로에 *_floor_input.txt와 *_floor_output.txt 관리.

입력

*_floor_input.txt 에서 read

```
1 12 12
1 1 3 1 1 1 1 1 1 1 1 1
1 2 2 2 2 2 2 2 2 2 2 1
1 2 1 1 2 1 1 1 1 1 2 1
1 2 1 2 2 1 2 2 2 1 2 1
1 2 1 2 1 2 2 1 2 1 2 1
1 2 1 6 1 2 1 1 2 1 1 1
1 2 1 1 1 2 2 1 2 1 2 1
1 2 1 2 1 1 2 1 2 1 2 1
1 2 2 2 2 2 2 1 2 2 2 1
1 1 1 1 1 1 1 1 1 1 2 1
1 2 2 2 2 2 2 2 2 2 2 1
1 1 4 1 1 1 1 1 1 1 1 1
```

출력

*_floor_output.txt 로 write

```
1 1 3 1 1 1 1 1 1 1 1 1
1 5 5 5 5 2 2 2 2 2 2 1
1 5 1 1 5 1 1 1 1 1 2 1
1 5 1 5 5 1 5 5 5 1 2 1
1 5 1 5 1 5 5 1 5 1 2 1
1 5 1 5 1 5 1 1 5 1 1 1
1 5 1 1 1 5 5 1 5 1 2 1
1 5 1 2 1 1 5 1 5 1 2 1
1 5 5 5 5 5 5 1 5 5 5 1
1 1 1 1 1 1 1 1 1 1 5 1
1 2 5 5 5 5 5 5 5 5 5 1
1 1 4 1 1 1 1 1 1 1 1 1
---
```

length=55
time=74

주의 사항

- 파일은 GitLab에 올릴 것!

- 경로 : (GitLab init 경로) – (assignment1) – [파일]

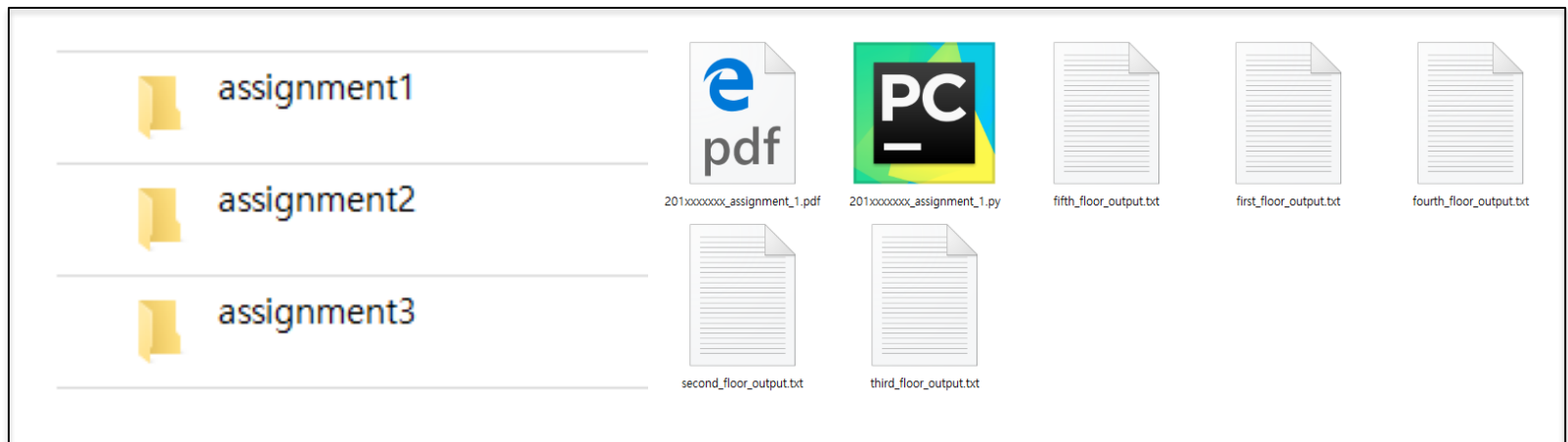
- 파일명 : ex)

본인학번_assignment_1.py, 본인학번_assignment_1.pdf,

***_floor_output.txt(5층 각각의 출력)**

프로젝트는 아래 그림과 같이 관리

- 소문자 assignment1, assignment2, ... 로 폴더를 만들어 과제 제출



Thank you!
