

[Python 트랙] 월말평가 – 알고리즘



| Background

- ✓ 이차원 배열
- ✓ 완전탐색
- ✓ DFS, BFS
- ✓ 컴퓨팅 사고력

| Goal

- ✓ 이차원 배열에 대한 단순 탐색을 할 수 있다.
- ✓ DFS 또는 BFS 를 활용하여 이차원 배열을 완전 탐색할 수 있다.
- ✓ 연산 복잡도에 대해 이해하고 식을 유도할 수 있다.

| 환경 설정

1) Pycharm(Pypy3.7 또는 Python3.7)을 이용해서 코드를 작성하고 결과를 확인한다.

- 새로운 Pycharm 프로젝트를 생성 후 코드를 작성한다.

2) 파일 이름 및 제출 방법

- 소스코드는 pypy에서의 한글 오류 방지를 위해 Algo1.py, Algo2.py로 작성한다.
- 디버깅이 끝나면 제출전에 파일이름을 다음 형식으로 바꾼다.

Algo1_서울_1반_이싸피.py

Algo2_서울_1반_이싸피.py

- 3번은 텍스트 파일로 작성한다.

Algo3_서울_1반_이싸피.txt

- 위 3개의 파일만 지역_반_이름.zip으로 압축하여 제출한다.

서울_1반_이싸피.zip

(탐색기에서 파일 선택 후 오른쪽 클릭 – 보내기 – 압축(zip)폴더 선택)

(edu.ssafy.com 사이트에 업로드)

3) 채점

- 주석이 없는 경우, 주석이 코드 내용과 맞지 않는 경우, 지정된 출력 형식을 만족하지 않는 경우 해당 문제는 0점 처리될 수 있다.

- import를 사용한 경우 해당 문제는 0점 처리될 수 있다. (import sys도 예외 없음)

4) 테스트케이스는 부분적으로 제공되며, 전체가 공개되지는 않는다.

5) 각 문제의 배점이 다르므로 표기된 배점을 반드시 확인한다.

- 1번 50점, 2번 30점, 3번 20점

성실과 신뢰로 테스트를 볼 것 (부정 행위시 강력 조치 및 근거가 남음)

※ 소스코드 유사도 판단 프로그램 기준 부정 행위로 판단될 시, 0점 처리 및 학사 기준에 의거 조치 실시 예정

[Python 트랙] 월말평가 – 알고리즘



| 문제 1 : 싸피 왕국의 당근 농장 (배점 50점)

싸피 왕국에는 $N \times N$ 크기의 당근 농장이 있고 1×1 크기의 구역들로 구성되어 있다. 지난밤 농장을 관리하는 싸린이가 없는 틈을 타서 M 마리의 토끼들이 당근을 훔쳤다.

토끼들은 당근을 훔칠 때 규칙에 따라서 당근을 훔쳐간다.

토끼가 당근을 훔칠 때의 규칙은 다음과 같다.

1. 토끼는 특정 위치부터 시작해서 당근을 훔친다.
2. 토끼는 상, 하, 좌, 우의 네 방향 중 한 방향으로만 이동 한다.
3. 토끼는 이동할 때 정해진 거리만큼 점프하면서 이동한다. 농장의 범위를 벗어나면 더 이상 당근을 훔치지 않는다.
4. 토끼는 자신이 방문하는 구역에서만 당근을 훔친다.
5. M 마리의 토끼들은 순서대로 당근을 훔치는 일을 한다.

예를 들어, 5×5 구역의 농장에서 4 마리의 토끼들이 당근을 훔치려고 한다. 4마리의 토끼들의 정보는 다음 표와 같다.

	시작 위치	방향	점프 거리
1번 토끼	(0, 1)	아래쪽	1
2번 토끼	(3, 1)	오른쪽	3
3번 토끼	(1, 3)	왼쪽	2
4번 토끼	(2, 2)	위쪽	2

1번 토끼가 (0, 1) 에서 아래로 1칸씩 점프하면서 당근을 훔친다. <그림1>에서 색칠된 구역이 토끼에 의해 피해를 입은 구역이다.

다음 2번 토끼가 (3, 1)에서 오른쪽으로 3칸씩 점프하면서 당근을 훔친다. <그림2>의 색칠된 구역이 1번과 2번 토끼에 의해 피해를 입은 구역이다. 2가 적힌 구역은 2번 토끼에 의해 피해를 입은 구역이다.

	0	1	2	3	4
0		1			
1		1			
2		1			
3		1			
4		1			

그림1

	0	1	2	3	4
0					
1					
2					
3		2			2
4					

그림2

[Python 트랙] 월말평가 – 알고리즘



3번 토끼가 (1, 3) 에서 왼쪽으로 2칸씩 점프하면서 당근을 훔친다.<그림3>

4번 토끼가 (2, 2) 에서 위쪽으로 2칸씩 점프하면서 당근을 훔친다.<그림4>.

	0	1	2	3	4
0					
1		3		3	
2					
3					
4					

그림3

	0	1	2	3	4
0			4		
1					
2			4		
3					
4					

그림4

토끼들에 의해 피해를 입은 구역은 색칠된 구역들이고, 총 **9** 개이다. 농장의 크기와 토끼들의 정보가 주어질 때 피해를 입은 구역의 수를 계산해보자.

[입력]

첫 줄에 테스트 케이스 개수 T가 주어진다. ($1 \leq T \leq 10$)

다음 줄부터 테스트 케이스별로 농장의 크기 N과 토끼 수 M이 주어진다.

($1 \leq N \leq 20$, $4 \leq M \leq 60$)

바로 다음 M개의 줄에 토끼의 시작 위치에 해당하는 행과 열의 값, 그리고 이동 방향, 점프하는 거리에 해당하는 4개의 정수 값들이 공백을 사이에 두고 주어진다.

토끼의 시작 위치에 해당하는 행과 열의 값은 0 에서 N-1 까지의 값이다.

방향 값은 0 에서 3까지 값으로 0부터 상(0),하(1),좌(2),우(3)를 나타낸다.

점프하는 거리는 1이상 3이하이다.

[출력]

#과 1번부터인 테스트케이스 번호를 출력하고, 토끼들에 의해 피해를 입은 구역의 수를 출력한다.

[Python 트랙] 월말평가 – 알고리즘



[입력 예시]

```

3          # tc 개수
5 4        # 1번 tc
0 1 1 1
3 1 3 3
1 3 2 2
2 2 0 2
5 5        # 2번 tc
1 1 1 2
0 3 1 1
2 4 2 3
3 0 3 1
4 3 0 2
5 15       # 3번 tc
0 1 2 1
4 3 2 1
2 3 2 2
3 2 3 1
2 2 3 1
0 2 3 2
1 0 2 1
1 3 1 2
3 4 2 2
1 4 0 2
0 3 3 2
4 2 3 2
0 0 2 3
3 0 2 2
3 1 3 2
    
```

[출력 예시]

```

#1 9
#2 12
#3 22
    
```

[Python 트랙] 월말평가 – 알고리즘



| 문제 2 : 어떤 땅을 사야 할까? (배점 30점)

김싸피는 전원생활을 꿈꾸며 땅을 사려고 한다. 최대한 넓은 땅을 싸게 사려고 한다.

김싸피가 사야하는 땅의 넓이와 가격을 구하는 프로그램을 작성하라.

N*N 형태의 격자안에 땅의 정보가 주어진다. 각 칸에는 땅의 가격이 주어지고, 0은 구매할 수 없는 칸이다. 또한 땅을 구매할 때는 연결되어 있는 땅은 모두 구입하여야 한다.

1	1	2	0	0	1	0
0	1	1	0	0	1	0
0	0	0	0	0	2	0
0	1	1	0	0	1	0
0	0	2	0	0	2	0
0	3	1	0	1	0	0
0	1	1	0	1	0	0

위 예시에서 가장 가장 넓은 땅은 녹색 땅이므로 7 10 을 출력한다.

[제약사항]

- N은 3이상 10이하 ($3 \leq N \leq 10$)
- 땅의 가격 K는 0 이상 5이하 정수 ($0 \leq K \leq 5$). 가격이 0인 땅은 구매할 수 없다.
- 넓이가 같은 경우, 가격이 더 낮은 땅을 구매한다.
- 연결된 땅이란 상하좌우 4방향으로 인접한 땅을 의미한다.

[Python 트랙] 월말평가 – 알고리즘



[입력]

첫 줄에 테스트케이스 수 T 가 주어진다.

각 테스트 케이스의 첫 줄에 N 이 주어진다.

이후 N 줄에 걸쳐 N 개의 정수가 띄어쓰기로 구분되어 주어진다.

[출력]

각 줄마다 " # T " (T 는 테스트 케이스 번호)를 출력한 뒤, 구매하는 땅의 넓이와 땅의 가격을 띄어쓰기로 구분해서 출력한다.

[입력 예시]

```
3
7
1 1 2 0 0 1 0
0 1 1 0 0 1 0
0 0 0 0 0 2 0
0 1 1 0 0 1 0
0 0 2 0 0 2 0
0 3 1 0 1 0 0
0 1 1 0 1 0 0
5
1 1 0 0 0
2 1 0 0 0
0 0 0 0 1
0 0 0 1 1
0 0 0 1 1
8
0 0 0 0 0 0 0 3
0 4 4 3 5 0 0 4
0 0 4 1 1 0 0 2
0 0 2 0 0 0 0 1
0 0 0 0 0 0 0 2
0 0 4 3 3 1 0 2
0 1 2 1 1 0 0 0
0 0 0 0 0 4 3 2
```

[출력 예시]

```
#1 7 10
#2 5 5
#3 8 16
```

[Python 트랙] 월말평가 – 알고리즘



| 문제 3 : 복잡도 (배점 20점)

1. 빅-오 표기법에 대해 설명하고, 복잡도가 $f(n) = 7n+4$ 이라 할 때 이를 빅-오(O) 표기법으로 표시하시오.

2. 다음 재귀식의 복잡도를 $O(\text{big-O})$ 표기 수준으로 풀어라. 4줄 이상의 풀이과정이 반드시 포함되어야 하고, $O()$ 표기법으로도 표기하시오.

$$T(n) = T(n - 1) + 1, T(0) = 1$$