



| Background

- ✓ 리스트와 반복문에 대한 이해
- ✓ 조합적 문제에 대한 이해
- ✓ 그래프 탐색과 최적화에 대한 이해

| Goal

- ✓ 반복문을 이용하여 리스트의 요소에 접근할 수 있다.
- ✓ 순열, 조합, 부분집합을 이용해 문제를 해결할 수 있다.
- ✓ 그래프 최적화 알고리즘을 이용해 문제를 해결할 수 있다.

| 환경 설정

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

- 새로운 Pycharm 프로젝트를 생성 후 코드를 작성한다.

2) 파일 이름 및 제출 방법

- 1, 2번 문제에 대한 소스 파일은 Algo문제번호_지역_반_이름.py로 만든다.

- pypy의 경우 파일이름에 한글을 사용할 수 없으므로 algo1.py, algo2.py 로 만들고 제출 시 아래와 같이 변경한다.

- 3번은 텍스트 파일로 작성한다.

Algo1_서울_1반_이싸피.py

Algo2_서울_1반_이싸피.py

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)

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 테스트에 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



| 문제 1 : 안전 지역 (배점 50점)

싸피 왕국과 파이썬 왕국은 오랜 시간에 전쟁을 해오고 있다. 파이썬 왕국은 싸피 왕국의 공격에 대비해 접전 지역에 방어탑을 설치한 상태이다.

방어탑은 상하좌우의 직선 방향으로 공격이 가능하다. 싸피 왕국의 정보국은 접전지역의 지도 정보를 입수했다.

정보국 소속인 싸린이는 방어탑의 공격을 피할 수 있는 안전 지역들을 찾아야 한다.

다음 <그림1> 은 5 x 5 지도 정보이다. 지도에 표시된 0은 빈 공간이고, 1은 장애물, 2는 방어탑이다. 방어탑의 공격은 장애물과 다른 방어탑을 통과 할 수 없다.

<그림2> 에서 노란색으로 칠해진 곳은 방어탑에서 공격가능한 지역들이다. 따라서, 안전지역의 개수는 총 7개가 된다.

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 2 | 2 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 2 | 1 | 2 | 0 |
| 0 | 0 | 0 | 0 | 0 |

<그림 1>

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 2 | 2 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 2 | 1 | 2 | 0 |
| 0 | 0 | 0 | 0 | 0 |

<그림 2>

N x N 크기의 지도 정보가 주어지면 안전 지역의 개수를 출력하는 프로그램을 작성하시오.



[입력]

- 첫 줄에 테스트케이스 수가 주어진다.
- 각 테스트케이스의 첫 줄에는 지도의 크기 $N(5 \leq N \leq 20)$ 이 주어진다
- 다음 N개의 줄에 공백으로 구분된 N개의 값이 주어진다.
- 0은 빈 공간, 1은 장애물, 2는 방어탑이다.

[출력]

- '#' 과 테스트 케이스 번호를 출력하고 공백을 두고 안전지역의 개수를 출력한다.

[입력예시]

```

3
5
0 1 0 0 0
0 0 2 2 0
0 1 0 0 0
0 2 1 2 0
0 0 0 0 0
5
0 0 0 0 0
0 2 1 2 0
0 1 0 1 0
0 2 1 2 0
0 0 0 0 0
5
0 0 1 0 0
1 2 0 1 0
0 1 2 0 1
1 0 0 2 0
0 1 2 0 1

```

(algo1_sample_in.txt 참고)

[출력예시]

```

#1 7
#2 9
#3 6

```

(algo1_sample_out.txt 참고)



| 문제 2 : 보석 수집 (배점 : 30점)

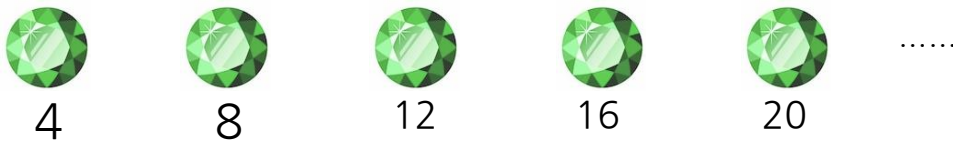
김싸피는 보석 수집을 하려고 한다. 예산에 맞게 보석을 수집할 때, 최대 가치를 출력하는 프로그램을 작성하시오.

보석은 아래와 같은 특징을 가진다.

- 김싸피가 모으는 보석은 다섯 종류이며, 보석의 번호는 4, 6, 7, 9, 11 이다.



- 각 보석은 보석 번호의 배수에 해당하는 가치를 가질 수 있다. 예를 들어 보석번호가 4인 경우 4, 8, 12, 16, 20 등의 가치를 가질 수 있다.



아래는 10개의 보석 중에서 예산 50에 맞게 보석을 고르는 예시이다. 10 개의 보석은 각각 4, 22, 1, 12, 11, 36, 21, 23, 3, 36 의 가치를 가지고 있다.

- 1, 23, 3인 보석은 김싸피가 모으는 보석 번호의 배수가 아니므로, 수집 대상이 아니다.
- 나머지 보석 중 4, 22, 12, 11 의 가치를 가지는 보석을 수집한다면, 총 49의 가치를 가지므로 예산안에서 최대 가치를 가진다.





[입력]

첫 줄에 테스트케이스 수가 주어진다.

각 테스트 케이스의 첫 줄에 보석의 개수 N 과 예산 M 이 띄어쓰기로 구분되어 주어진다.

두 번째 줄에서 N 개의 정수 A_i ($0 \leq i < N$) 가 띄어 쓰기로 구분되어 주어진다.

($3 \leq N \leq 20, 0 \leq M \leq 200, 1 \leq A_i \leq 100$)

[출력]

각 줄마다 " #T " (T는 테스트 케이스 번호)를 출력한 뒤, 최대 보석 가치를 출력한다.

[입력 예시]

```
3
10 50
4 22 1 12 11 36 21 23 3 36
11 50
4 22 1 12 11 6 21 23 3 36 6
10 50
3 2 5 5 1 13 23 17 50 25
```

(algo2_sample_in.txt 참고)

[출력 예시]

```
#1 49
#2 50
#3 0
```

(algo2_sample_out.txt 참고)



| 문제 3 : 최소 신장 트리 (20점)

- 1) 최소 신장 트리(Minimum Spanning Tree)에 대해 간단히 설명하시오
- 2) Kruskal 알고리즘의 동작에 대해 설명하시오. (슈도코드 없이 설명해도 무방함)