



[Python트랙] 월말평가2 - 알고리즘 기본

| Background

- ✓ 선형 자료구조에 대한 이해와 활용
- ✓ 완전 탐색 및 백트래킹에 대한 이해와 활용
- ✓ 트리에 대한 이해와 활용

| Goal

- ✓ 배열과 스택, 큐, 트리의 특징에 대해 설명할 수 있다.
- ✓ 스택과 큐를 사용해 문제를 해결할 수 있다.
- ✓ 완전탐색과 백트래킹으로 문제를 해결할 수 있다.

| 환경 설정

1) Pycharm과 pypy 또는 python을 이용해서 코드를 작성하고 결과를 확인한다. **새로운 프로젝트를 생성하지 않고 기존 프로젝트 사용시 부정행위로 간주 함.**

2) 파일 이름 및 제출 방법

- 1, 2번 문제에 대한 소스 파일 이름은 다음과 같이 영문으로 작성한다.

서울 1반 이싸피라면, algo문제번호_반_이름.py 순서로 영문으로 작성

algo1_01_leessafy.py

algo2_01_leessafy.py

- 3번 문제에 대한 답안 파일 이름은 .txt 형식으로 다음과 같이 영문으로 작성한다.

algo3_01_leessafy.txt

- 위 3개의 파일만 지역_반_이름.zip으로 압축하여 제출한다.

서울_1반_이싸피.zip

(탐색기에서 파일 선택 후 오른쪽 클릭 – 압축대상 – Zip 선택)

3) 채점

- 문제별로 부분 점수가 부여된다.
- 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 40점, 2번 35점, 3번 25점

성실과 신뢰로 테스트에 임할 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**

[Python트랙] 월말평가2 - 알고리즘 기본



| 문제1 : 미생물 이동 (배점 : 40점)

김싸피가 실험중인 미생물은 N개의 활동 구역을 왼쪽에서 오른 쪽으로 이동하고 있다. 미생물은 어떤 구역에서 먹이를 먹으면 오른쪽으로 K칸 만큼 더 이동할 수 있고, 추가로 먹이를 먹지 못하면, K번째 칸에 멈추게 된다고 한다.



먹이가 있는 활동구역에 대한 정보가 주어지고, 미생물이 가장 왼쪽 칸에 놓여지면, **미생물이 최대 몇 번째 칸까지 이동할 수 있는지** 알아내는 프로그램을 만드시오.

먹이가 있으면 1, 없으면 0으로 표시된 N개의 활동 구역과 미생물이 먹이를 먹고 이동하는 최대 칸 수 K가 제공된다. 첫 번째 구역에는 항상 먹이가 있다.

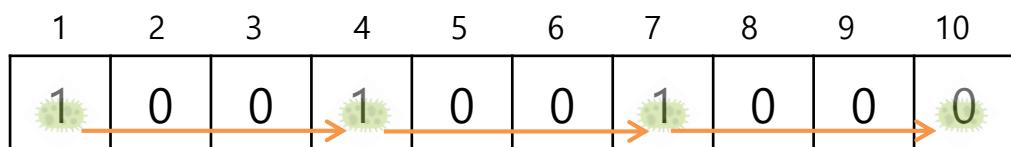


그림1. $N = 10, K = 3$ 인 경우

그림1의 경우, $N = 10, K = 3$ 인 예로, 미생물이 먹이를 먹을 때마다 3칸씩 이동할 수 있으므로 이 경우 미생물은 10번째 칸까지 이동할 수 있다.

(다음 페이지에 계속)

[Python트랙] 월말평가2 - 알고리즘 기본



1	0	0	1	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---

그림2. $N = 10$, $K = 3$ 인 경우

먹이 배치가 그림2와 같은 경우, 9번째 칸의 먹이를 먹어도 활동구역을 벗어날 수는 없고 10번째 칸에 멈추게 된다. 따라서 최대 이동 위치는 10번째 구역이다.

1	0	0	1	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---

그림3. $N = 10$, $K = 2$ 인 경우

그림3의 경우, 첫 번째 칸에서 먹이를 먹은 미생물은 3번째 칸에서 멈추게 된다.

1	1	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---

그림4. $N = 10$, $K = 3$ 인 경우

그림4의 경우, 미생물은 5번째 칸에서 멈추게 된다.

(다음 페이지에 계속)

[Python트랙] 월말평가2 - 알고리즘 기본



[입력]

첫 줄에 활동 구역의 수 T가 주어지고, 활동 구역별로 첫 줄에 N과 K, 다음 줄에 N개 구역의 막이 정보가 빈칸으로 구분되어 주어진다.
($3 \leq T \leq 10$, $1 \leq N \leq 10$, $K \leq 10$)

[출력]

각 활동 구역별로 한 줄에 #과 1번 부터인 구역번호, 빈칸에 이어 최대로 몇 번째 구역에 이동할 수 있는지 출력한다.

[입력 예시]

```
3
10 3
1 0 0 1 0 0 1 0 0 0
10 3
1 0 0 1 0 0 1 0 1 0
10 2
1 0 0 1 0 0 1 0 0 0
```

(algo1_sample_in.txt 참고)

[출력 예시]

```
#1 10
#2 10
#3 3
```

(algo1_sample_out.txt 참고)

[Python트랙] 월말평가2 - 알고리즘 기본



| 문제2 : 완만한 오르막 (배점 : 35점)

* 내장함수 max(), min(), abs() 사용금지

김싸피는 등산을 좋아해 주말마다 등산을 한다. 하지만 체력이 약한 김싸피는 최대한 완만한 경사도를 가지는 오르막 길을 좋아한다. 전체 경로 길이가 N인 경로에서 김싸피가 제일 좋아하는 오르막 길의 길이를 출력하는 프로그램을 작성하시오.



1	3	4	5	4	6	7	7	6	9
---	---	---	---	---	---	---	---	---	---

- 등산 경로의 지형은 정수 배열 형태로 주어진다.
- 오르막 길은 이전 위치 보다 **높거나 같은 경우** 계속 이어진다고 판단한다.
- 경사도는 오르막길에서 가장 높은 높이 A 와 가장 낮은 높이 B의 차에서 경로 길이 C 를 나눈 값이다. 예) 오르막길 경로가 [1, 3, 4, 5] 인 경우, A = 5, B = 1, C = 4 이므로 $(A - B) / C = 1$, 따라서 경사도는 1이다.
- 만약 경사도가 같은 경우, 오르막 길이가 **긴 경로의 길이**를 출력한다.
(경사도가 같은 경우에 대해 부동소수점 수의 오차는 고려하지 않아도 된다.)
- **최소** 오르막 길이는 2 이다.

예시) 다음은 N = 10 인 등산경로 예시이다.

1	3	4	5	4	6	7	7	6	9
---	---	---	---	---	---	---	---	---	---

- 주어진 예시에서는 총 3개의 오르막 경로가 존재한다.
- (1) [1,3,4,5] 경사도 1 (2) [4,6,7,7] 경사도 0.75 (3) [6,9] 경사도 1.5
위 세 개의 경로 중 경사도가 가장 낮은 경로는 경사도 0.75 로 (2)번 경로가 가장 경사도가 완만하다. 따라서 프로그램은 2번 경로의 길이인 4를 출력해야 한다.

[Python트랙] 월말평가2 - 알고리즘 기본



입력

첫 줄에 테스트 케이스 개수 T가 주어진다.

각 테스트 케이스의 첫 줄에는 N이 주어진다.

다음 줄에 N개의 자연수 A_i 가 띄어쓰기로 구분되어 주어진다.

$(5 \leq N \leq 20, 1 \leq A_i \leq 100)$

출력

각 줄에 테스트케이스 번호를 #번호 형태로 출력하고, 한 칸 띄워 김싸피가 가장 좋아하는 경로 길이를 출력한다. 만약 오르막이 없는 경우 0을 출력한다.

[입력 예시]

```
3
10
1 3 4 5 4 6 7 7 6 9
10
1 4 7 3 4 5 7 11 13 17
6
1 2 7 2 6 1
```

(algo2_sample_in.txt 참고)

[출력 예시]

```
#1 4
#2 7
#3 3
```

(algo2_sample_out.txt 참고)

[Python트랙] 월말평가2 - 알고리즘 기본



| 문제3 : 서술형 (배점 : 25점)

3-1. 스택과 큐의 특징에 대해 삽입과 삭제 순서를 중심으로 간단히 설명하시오.

3-2. DFS와 백트래킹의 차이에 대해 간단히 설명하시오.