

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용

## | Background

- ✓ 프로그래밍의 논리적인 이해
- ✓ 다양한 알고리즘 기법과 적용에 대한 이해.

## | Goal

- ✓ 문제를 분석하고 알맞은 알고리즘 기법을 적용하여 자바로 코드화 할 수 있다.

## | 환경 설정

1) 워크스페이스 변경 : 이클립스 시작 시 워크스페이스는 **c:\WAlgoTest9** 으로 변경합니다.

(이클립스 시작 시 워크스페이스를 물어보는 창이 보이지 않고 바로 이클립스가 실행될 경우 File> Switch Workspace > Other 메뉴를 이용하여 워크스페이스를 변경합니다.)

2) 인코딩 (utf-8)

이클립스 메뉴-> Window -> Preferences -> General -> Workspace : UTF-8 로 변경

3) 프로젝트 생성 : 적당한 이름(Algo)으로 프로젝트를 생성한다.

4) 소스코드 작성- 아래 3개의 이름으로 main()를 포함한 클래스와 텍스트 파일을 생성한다. 소스코드는 라인별로 소스코드에 대한 주석을 작성한다.

(단, src 폴더 아래 패키지 없이 작성-default package)

**Algo1\_도시\_반\_이름.java , Algo2\_도시\_반\_이름.java , Algo3\_도시\_반\_이름.txt**

(도시: 서울, 구미, 대전, 광주, 부울경)      예시> Algo1\_서울\_08반\_홍길동.java

5) 제출 파일 : 위 3개의 소스파일과 텍스트 파일만 **Algo 도시 반 이름.zip** 으로 압축하여 제출한다. (edu.ssafy.com 사이트에 업로드)

6) 테스트 케이스 : 모든 테스트 케이스는 공개되지 않으며, 부분적으로 제공됩니다.

7) 채점 : 테스트 케이스별로 부분 채점 된다.

## 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

## [자바전공 트랙] 9회차 과목평가\_알고리즘응용

## | 문제1. 로봇 경주장[40점] (Algo1\_도시\_반\_이름.java)

경주를 하는 로봇은 토끼를 모티브로 만들어졌다. 그래서 토끼 로봇은 앞다리에 비해 뒷다리가 길고 튼튼하다. 뒷다리가 길어 높이 뛰어오를 수 있으며 오르막도 곧잘 오르지만 내리막은 잘 달리지 못한다.



싸피 강의에 적극 참여하여 모범이 된 교육생인 김싸피는, 머리를 식힐 겸 토끼 로봇이 경주를 벌이는 로봇 경주장을 방문했다. 김싸피는 5 종류의 토끼 로봇 중 얼마나 많은 토끼 로봇이 경주장을 완주할 수 있는지 알아보려고 한다. 경주장은 오르막길과 내리막길이 있으며, 각 지점마다 높이가 정해진다. 경주장의 각 지점의 높이  $H_i$  는 0보다 크거나 같고 100보다 작거나 같다. 경주장의 길이는 10이며 아래는 경주장 샘플이다.

0 20 50 70 50 60 40 50 30 0

이 샘플 경주장에서 1번째 지점에서 2번째 지점으로의 오르막은 20미터, 2번째 지점에서 3번째 지점으로의 오르막은 30미터, 그리고 9번째 지점에서 10번째 지점으로의 내리막은 30미터이다.

경주장을 뛸 5 종류 토끼 로봇은 오르막 점프 한계치와 내리막 점프 한계치를 가지며, 자신이 가진 한계치보다 높은 오르막 혹은 내리막은 이동하지 못한다. 5종류의 토끼 로봇 중 몇 개의 토끼 로봇이 경주장을 완주할 수 있을지 검사하는 프로그램을 작성하시오.

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용



## | 문제1. 로봇 경주장[40점] (Algo1\_도시\_반\_이름.java)

[입력]

첫 번째 줄에 테스트케이스의 개수가 주어지고,

각 테스트케이스의 첫번째 줄에는 경주장 10개의 지점의 높이  $H_i$ 가 주어진다.

(  $0 \leq H_i \leq 100$  )

각 테스트케이스의 두 번째 줄부터 여섯 번째 줄 까지 각 줄마다 5마리의 토끼의 오르막 한계치 A, 내리막 한계치 B가 공백을 두고 입력된다. (  $0 \leq B \leq A \leq 100$  )

[출력]

테스트케이스 별로 #T (T는 테스트케이스의 번호)를 출력한 후 한 칸 띄워 경주장을 완주할 수 있는 토끼의 수를 출력한다.

[제약사항]

시간제한 1초      메모리제한 256MB

[예제 입력]

```
2
0 20 50 70 50 60 40 50 30 0
30 20
60 40
50 30
40 40
100 0
100 90 90 90 90 90 90 90 90 50
```

```
30 20
60 40
50 30
40 40
100 0
```

[출력 예시]

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

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용

## | 문제2. 기둥 옮기기[30점] (Algo2\_도시\_반\_이름.java)

가로와 세로의 길이가 같은 평지에서 기둥을 옮기려 한다. 그 지형은 0과 1로 나타나 있다. 1은 세워져 있는 기둥을 나타내고 0은 아무 것도 없음을 나타낸다. 다음 지형을 보자.

B	0	0	1	1
B	0	0	0	0
B	0	0	0	0
1	1	0	0	0
E	E	E	0	0

위의 지형에서 길이 3인 기둥 BBB를 밀거나 회전시켜 EEE의 위치로 옮기는 작업을 하는 문제를 생각해 보자. BBB와 EEE의 위치는 임의로 주어진다. 단 문제에서 기둥의 길이는 항상 3이며 B의 개수와 E의 개수는 같다. 기둥을 움직이는 방법은 아래와 같이 상하좌우(Up, Down, Left, Right)와 회전(Turn)이 있다.

코드	의미
U	기둥을 위로 한 칸 옮긴다.
D	기둥을 아래로 한 칸 옮긴다.
L	기둥을 왼쪽으로 한 칸 옮긴다.
R	기둥을 오른쪽으로 한 칸 옮긴다.
T	중심점을 중심으로 90도 회전시킨다.

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용

## | 문제2. 기둥 옮기기[30점] (Algo2\_도시\_반\_이름.java)

예를 들면, 다음과 같다. (초기상태로부터의 이동)

초기상태	상( U )	하( D )
<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 B B B 0 0 0 0 0 0 0 0 0 0 0 1 0 0 </pre>	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 B B B 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 </pre>	<pre> 0 B B B 0 0 0 0 0 1 0 0 </pre>
좌( L )	우( R )	회전( T )
<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 B B B 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 </pre>	<pre> 0 B B B 0 0 0 0 0 0 0 0 0 0 1 0 0 </pre>	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 B 0 0 0 0 0 B 0 0 0 0 0 B 0 0 0 0 0 0 1 0 0 </pre>

이와 같은 방식으로 이동시킬 때에 그 움직일 위치에 다른 기둥, 즉 1이 없어야만 움직일 수 있다. 그리고 움직임은 위의 그림과 같이 한 번에 한 칸씩만 움직인다. 단 움직이는 기둥은 어떤 경우이든지 중간단계에서 한 행이나 한 열에만 놓일 수 있다. 예를 들면 아래 그림에서 a와 같은 단계는 불가능하다. 그리고 회전의 경우에는 반드시 중심점을 중심으로 90도 회전을 해야 한다. (항상 기둥의 길이가 3이므로 중심점이 있음)

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용

## | 문제2. 기둥 옮기기[30점] (Algo2\_도시\_반\_이름.java)

그리고 이런 회전(Turn)이 가능하기 위해서는 그 통나무를 둘러싸고 있는 3\*3 정사각형의 구역에 단 하나의 기둥도 없어야만 한다. 즉, 아래그림 b, d와 같이 ?로 표시된 지역에 다른 기둥, 즉 1이 없어야만 회전시킬 수 있다. 따라서 c와 같은 경우에, 기둥은 왼쪽에 아직 철거되지 않은 기둥 때문에 회전시킬 수 없다.

a	b	c	d
<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 B 0 0 0 0 B 0 0 0 0 B 0 0 0 0 0 0 0 1 0 0                     </pre>	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ? ? ? 0 0 0 B B B 0 0 0 ? ? ? 0 0 0 0 0 0 0                     </pre>	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 B 0 0 0 0 0 B 0 0 0 0 0 B 0 0 0 0 0 0 0 0                     </pre>	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ? B ? 0 0 0 ? B ? 0 0 0 ? B ? 0 0 0 0 0 0 0                     </pre>

문제는 기둥을 5개의 기본동작(U, D, L, R, T)만을 사용하여 처음위치(BBB)에서 최종위치(EEE)로 옮기는 프로그램을 작성하는 것이다. 단, 최소 횟수의 단위 동작을 사용해야 한다.

### [입력]

첫째 줄에 주어진 평지의 한 변의 길이 N이 주어진다. ( $4 \leq N \leq 50$ ) 주어진다. 이어서 그 지형의 정보가 0, 1, B, E로 이루어진 문자열로 주어진다. 한 줄에 입력되는 문자열의 길이는 N이며 입력 문자 사이에는 빈칸이 없다. 기둥과 최종 위치의 개수는 1개이다.

### [출력]

첫째 줄에 최소 동작 횟수를 출력한다. 이동이 불가능하면 0만을 출력한다.

- ❖ 사용 가능한 언어      Java
- ❖ 제한시간                2초
- ❖ 메모리제한             128MB

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용



## | 문제2. 기둥 옮기기[30점] (Algo2\_도시\_반\_이름.java)

❖ 예제 입력 1

5

B0011

B0000

B0000

11000

EEEE00

❖ 예제 출력 1

9

❖ 예제 입력 2

4

B000

B01E

B00E

000E

❖ 예제 출력 2

0

# [자바전공 트랙] 9회차 과목평가\_알고리즘응용



## | 문제3. 컴퓨팅 사고력[30점] (Algo3\_도시\_반\_이름.txt)

- 1)  $A = 10009999 \times 10009999 \times 10009999 \times 10009999$  일 때,  
A를 8로 나누면 나머지는 얼마인지 적으시오.
- 2)  $A = 2023 \times 2023 \times 2023 - 2021$ 을 6으로 나누면 나머지는 얼마인지 적으시오.
- 3)  $2 \times n$  직사각형을  $1 \times 2$ ,  $2 \times 1$ 과  $2 \times 2$  타일로 채우는 방법의 수를  $f(n)$ 이라 할 때  $f(7)$ 를 구하시오. (단  $f(1)=1$ ,  $f(2)=3$ 이다)