

# 자료 구조 HW #1 (30점)

제출일 : 2017년 4월 19일 수요일 (eCampus)

제출물 : 학번.java 파일 하나

Hard copy 제출 필요 없음

hw1.zip 파일 : LabTest.java hw1.java lab.in lab.out hw1.pdf

---

## 제출

hw1.java 를 학번.java 로 변경하여 이 파일 한 개만 제출할 것.

---

이번 숙제는 미로 찾기 알고리즘을 구현하는 내용이다. 사용자로부터 미로의 내용을 입력 받고, 이를 바탕으로 최종 시작점까지 찾아가는 내용이다.

수행 예는 다음과 같다.



```
kmucs@localhost: ~/dbox/classes171/ds/hw/hw171
kmucs@localhost:~/dbox/classes171/ds/hw/hw171$ java LabTest
Maze >
maze
Enter the dimension m, p of the maze
4 4
Enter the Maze
0 1 0 0
0 1 0 0
1 0 0 1
1 1 0 0
(1,1)
(2,1)
(3,2)
(3,3)
(4,4)
Maze >
quit
kmucs@localhost:~/dbox/classes171/ds/hw/hw171$
```

우선 사용자로부터 미로의 크기를 입력 받는다. 이를 위해 "maze"라는 명령어를 입력한다.

그러면 프로그램은  $m$ 과  $p$ 를 입력하라고 한다.  $m$ 은 행 개수를 의미하고,  $p$ 는 열 개수를 의미한다. 그 다음의 미로의 내용을 입력하는데 교과서에 나온 알고리즘 대로 막힌 곳은 1로 표시하고, 열린 곳은 0으로 표시한다.

미로의 입력이 끝나면 경로를 탐색하여 그 결과를 출력한다. 입력 받은 미로의 맨 왼쪽 맨 위 행이  $(1,1)$ 에 해당한다.

경로를 탐색하는 과정은 수업시간에 설명된 내용 그대로인데, 단 한가지 차이점은 여덟 방향을 탐색할 때 탐색하는 순서는 다음과 같다. **이 부분은 매우 중요함.**

E, SE, NE, N, NW, W, SW, S

따라서 위의 예제의 경우  $(3,2)$  다음에  $(2,3)$ 이나  $(4,3)$ 으로 가지 않고,  $(3,3)$ 으로 간다.

그리고 탐색의 시작은 항상  $(1,1)$ 위치에서 시작하고, 탐색의 끝은  $(m, p)$  위치에서 끝난다.

참고로, 이번 숙제에서 구현할 알고리즘이 최적의 알고리즘은 아니므로 결과 확인 시 최적이지 아니더라도 고민하지 말 것.

이 알고리즘을 구현하기 위해 Java의 Stack 클래스를 사용해도 된다.

<https://docs.oracle.com/javase/7/docs/api/java/util/Stack.html>

이중 Method Summary 란에 있는 method 들 (empty, pop, push) 을 사용하면 도움이 됨.

프로그램의 이해와 테스트를 돕기 위해 hw1.zip에는 다음과 같은 파일이 포함되어 있다.

lab.in : 사용자가 입력할 수 있는 예제들이다.

lab.out : hw1.in의 입력 시 출력되는 내용이다. 아래의 프로그램 테스트에 사용된다.

## 프로그램 테스트

### 컴파일

```
$ javac hw1.java LabTest.java
```

### 실행

```
$ java LabTest
```

주어진 **input**으로 실행

```
$ java LabTest < lab.in
```

주어진 **output**과 비교

```
$ java LabTest < lab.in > abc
```

```
$ diff abc lab.out
```

또는

```
$ diff -i --strip-trailing-cr -w abc lab.out
```