

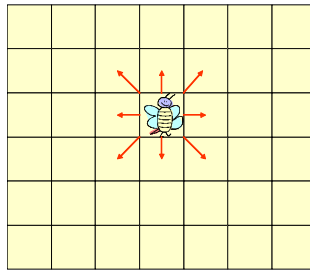
# 2016 C++ 프로그래밍 실습과제 9

※ Random Walk 문제를 2차원 배열을 이용한 행렬 클래스를 이용해 구현하시오.

## 9.1 수학에서의 "random walk"라 불리는 문제 구현

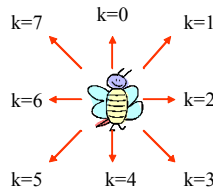
(1) 문제 설명:

술에 취한 딱정벌레가  $n \times m$  크기의 타일로 구성된 방안에 있다. 딱정벌레는 임의의(랜덤) 위치를 선택하여 여기저기 걸어 다닌다. 현재의 위치에서 주위의 8개의 타일로 걸어가는 확률은 동일하다고 가정하자. 그러면 최종적인 문제는 "딱정벌레가 방안의 모든 타일을 한 번씩 지나가는데 몇 번의 이동이 필요한가?"



방 전체를 강의자료의 행렬 클래스를 이용하여 2차원 배열로 모델링한다.

- 처음에는 딱정벌레가 배열의 중앙에 있다고 가정하라.
- map의 초기 값은 0이다.
- 딱정벌레가 타일을 지나갈 때마다 2차원 배열의 값을 1로 만들어서 딱정벌레가 지나갔음을 나타낸다.
- 0부터 7까지의 랜덤한 숫자를 생성하여 다음 그림과 같이 움직인다. 즉 0이면 북쪽으로 이동하고 4이면 남쪽으로 이동한다. (순서는 달라도 됨)



0부터 7까지의 랜덤한 숫자는 다음과 같이 생성할 수 있다. 즉 rand() 함수의 반환 값을 8로 나누어 나머지를 취한다.

```
number = rand() % 8;
```

- 모든 타일값을 검사하여 전체 타일이 1이 되면 프로그램을 종료한다.

(3) 강의자료 8장의 행렬 클래스에 대한 UML 다이어그램을 작성하라.

- 행렬 클래스(MatrixI)의 private 멤버를 RandomWalk에서 접근할 수 있도록 다음과 같은 문장을 MatrixI 클래스에 추가한다.

```
friend class RandomWalk;
```

<MatrixI 클래스 일부>

```
class MatrixI {
    int    rows;    // 행의 개수
    int    cols;    // 열의 개수
    int**  mat;     // 행렬 데이터
    friend class RandomWalk;

public:
    MatrixI( int r=0, int c=0 ): mat(NULL) { alloc(r,c); }
    ~MatrixI( ) { reset(); }
```

(4) 다음과 같은 RandomWalk 클래스에 대한 UML 다이어그램을 작성하라.

<RandomWalk 클래스>

```
#pragma once
#include "matrix.h"
#include <time.h>
#include <windows.h>

// Random Walk를 위한 전역변수를 설정하는 함수
// 입력: 행의 수와 열의 수
// 2차원 배열 _map을 동적으로 할당하고 초기화 함
class RandomWalk {
    MatrixI    map;           // 맵
    int        nMove;         // 현재 이동 횟수
    int        x;             // 현재 위치
    int        y;             // 현재 위치

public:
    RandomWalk( int rows=0, int cols=0 ): map(rows, cols), nMove(0) {}

    int    Rows()              { return map.rows; }
    int    Cols()              { return map.cols; }
    int&   Map(int i, int j)   { return map.mat[i][j]; }

    // 별레의 초기 위치를 설정
    void setPosition ( int yy, int xx ) { x = xx; y = yy; }

    // 별레가 Random walk map의 모든 부분을 지나갔는지를 판단
    bool isDone ( ) { ... }

    // Random walk map의 현재 상태를 화면에 출력함
    void print ( ) { ... }

    // 별레가 현재 위치에서 무작위로 8방향 이웃중 하나로 이동하는 함수
    // 별레가 map의 경계 밖으로 나가지 못하도록 처리해야 함
    void move( ) { ... }

    // Random Walk의 메인 함수
    // 입력: 최대 이동 횟수(maxwalk)와 한번 이동 후 쉬는 시간(wait)를 입력
    // 쉬는 시간을 위해 Sleep(msec) 함수 이용 (사용법은 인터넷 찾아볼 것)
    void play ( int maxwalk, int wait=0 ) { ... }
};
```

(5) 첨부된 실행 파일과 같이 동작하도록 각 클래스를 구현하고 다음과 같은 `main()` 함수에서 동작을 테스트하라.

<main()함수>

```
#include "RandomWalk.h"

void main()
{
    int r, c, maxwalk;

    srand( (unsigned int)time(NULL) );

    printf(" < 술취한 딱정벌레 문제 >\n\n");
    printf(" 배열의 크기를 입력하세요: ");
    scanf("%d%d", &r, &c);

    printf(" 최대 움직임 횟수를 입력하세요: ");
    scanf("%d", &maxwalk);

    RandomWalk game(r,c);
    game.play(maxwalk);
}
```