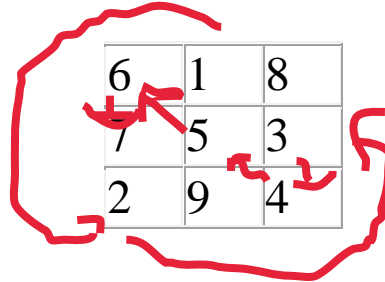


Lab2: Magic Square problem

magic square란, 1 에서 n^2 까지의 정수로된 $n \times n$ 행렬로서, 각행의 합, 열의 합, 주 대각선의 합이 모두 같다. 또한 크기가 n 이고, n 은 홀수 이다.

Ex) Magic Square 3: 열, 행 및 주 대각선의 합이 15



6	1	8
7	5	3
2	9	4

● 알고리즘

```
procedure magic() {  
    int square[max] [max], k, l;  
  
    // square 를 0 으로 초기화    // n = size of the square  
    .....  
    // i 와 j 는 현재 위치  
    int key = 2;    // key 값은 2 부터 n x n 까지 1 씩 증가  
    i = 0;    int j = n / 2;  
    square[i][j] = 1;    // 첫 행의 중간에서 시작.  
  
    while(key <= n * n) {    // 왼쪽 위방향, 대각선으로 이동  
        if(i - 1 < 0) row = n - 1;    else row = i - 1;  
        if(j - 1 < 0) col = n - 1;    else col = j - 1;  
  
        if(square[row][col])    i = (i + 1) % n;  
                                //square full, move down  
        else { i = row;    j = col;}  
    }  
    square[i][j] = key++;  
} // end while
```

if key =3
key < 3*3

```
// 매직 스퀘어를 출력
.....
}
```

조건:

- 1) 데이터 n: 키보드 입력: 예) 크기가 5 인 매직스퀘어
- 2) 첫 번째 행의 중앙에 1 을 넣는다.
- 3) 출력은 크기가 5 일 경우 아래 테이블처럼 출력할 것

15	8	1	24	17
16	14	7	5	23
22	20	13	6	4
3	21	19	12	10
9	2	25	18	11

<< Extra Points>> 다항식의 덧셈 $A(X) + B(X) = C(X)$

1) 다항식의 표현: typedef struct { float coef; int expon; } Polynomial;

2) 테스트 데이터

$$A(X) : 3x^4 + 5x^2 + 6x + 4$$

3	5	6	4
4	2	1	0

$$B(X) : 5x^3 + 4x^2 - 6x + 1$$

5	4	-6	1
3	2	1	0

3) 출력: $C(X) : 3x^4 + 5x^3 + 9x^2 + 5$

3	5	9	5
4	3	2	0