

달팽이 과제

```
기타 파일 (전역 범위) main()
1  #include <stdio.h>
2
3  int main()
4  {
5      int input, m, l = 0, k = 1, num = 1;
6      int arr[10][10];
7      int sum = 0, arr1 = 0;
8      scanf_s("%d", &input);
9  }
```

배열 크기가 [10][10]인 이차원 배열 arr 만들어주고
Sum이랑 arr1은 합계 구할 때 쓰기 위해 선언했습니다

input == 2

0,0	0,1	1	2
1,0	1,1	4	3

input == 3

0,0	0,1	0,2	1	2	3
1,0	1,1	1,2	8	9	4
2,0	2,1	2,2	7	6	5

input == 4

0,0	0,1	0,2	0,3	1	2	3	4	0,-	-,3	3,-
1,0	1,1	1,2	1,3	12	13	14	5	-,0	1,-	-,2
2,0	2,1	2,2	2,3	11	16	15	6			
3,0	3,1	3,2	3,3	10	9	8	7	2,-		

input == 5

0,0	0,1	0,2	0,3	0,4	1	2	3	4	5
1,0	1,1	1,2	1,3	1,4	16	17	18	19	6
2,0	2,1	2,2	2,3	2,4	15	24	25	20	7
3,0	3,1	3,2	3,3	3,4	14	23	22	21	8
4,0	4,1	4,2	4,3	4,4	13	12	11	10	9

0,- -,4 4,- -,0

1,- -,3 3,- -,1

2,-

Input == 5 아래에 숫자들 보시면
규칙이 있어요
그래서 포문을 4개를 이용해서 달팽이를
만들기로 했어요

```

for (int i = 0; i < input; i++)
{
    for (int j = k - 1; j < input - k + 1; j++)
    {
        arr[i][j] = num;
        num++;

        if (num > input * input)
        {
            continue;
        }
    }
}

```

포문 안에서 int i로 초기화한 이유는
포문 안에서 선언하면 지역변수로 선언돼서 메모리
덜 차지해서 더 효율적이라고
선배님이 알려주셨습니다

저 아래에 if 문은 num이 배열에 들어가는 숫자인데
배열 안에서 최대치가 input제곱 값인데 num이 그거
넘어서 계속 커지면 안되니까 그거 막으려고 했습니
다

<i 포문 역할>

i로 반복문 크게 돌리고
첫번째 포문은 오른쪽으로 숫자 커지는 역
할이에요

<k 정의>

K는 1로 해놓고 첫번째 완전 큰 포문이 끝
나면 k++해서 1씩 커지게 해줬어요

j를 k-1로 한 이유가

큰 포문 1번 돌면 (1,1)부터 시작 되어야 해
요. 그래서 j가 i 포문 돌때마다 1씩 커져야
하니까 변수로 초기화 해줬습니다

Input - k + 1로 한 이유는 큰 포문 돌 때마
다 점점 작은 곳에서 멈춰야하니까
K라는 점점 커지는 변수만큼 빼주고 1 더해
줬습니다

```

}
for (int j = k; j <= input - k; j++)
{
    arr[j][input - 1 - i] = num;
    num++;

    if (num > input * input)
    {
        continue;
    }
}

```

오른쪽 가장자리에서 아래로 내려가는 역할하는 포문이에요

J가 k로 초기화되어있는 이유는 k는 첫번째 포문 도는 과정이라 했을 때 k 값이 1이니까

4번째 페이지 포문에서 첫번째줄은 num이 다 들어갈 있을 거잖아요

그래서 그 다음칸부터 num이 들어가야하니까 k로 해줬고

큰 포문 돌때마다 점점 한줄씩 밀려나야해요

위에 4번째 페이지 포문이 한줄을 다 num 들어가게 하니까

그래서 1씩 커지는 k로 해줬고

달팽이 돌아갈 수록 점점 안에 공간이 작아져야하잖아요 그래서 input - k로 점점 작아지게끔 해줬어요

[][input-1-i]로 한 이유는 배열 보시면 숫자가 고정되어 있어요 근데 큰 포문 돌 때마다 작아져야 해요

그래서 이렇게 해줬습니다

```

}
for (int j = input - k - 1; j >= i; j--)
{
    arr[input - 1 - i][j] = num;
    num++;

    if (num > input * input)
    {
        continue;
    }
}
for (int i = input - k - 1; i >= k; i--)

```

이 포문은 맨 아래에서 왼쪽으로 숫자가 커지는 역할인데요

보시면 `arr[j]`에서 `j`가 `input-k-1`에서 1씩 작아지잖아요

저렇게 작아지게 하면서 `num` 키워서 왼쪽으로 숫자 커지게 해줬습니다

`arr[input-1-i]`는 배열보시면 인덱스에서 1 뺀 만큼으로 다 되어있어요 근데 큰 포문 돌면 1 더 작아져야해요 그래서 `i`만큼 빼줬어요

근데 `k`로 해줘도 좋을 거 같아요

`Input-k`로

```

    }
    for (int j = input - k - 1; j >= k; j--)
    {
        arr[j][i] = num;
        num++;

        if (num > input * input)
        {
            continue;
        }
    }
    k++;
}

```

이걸 왼쪽 가장자리에서 올라가는 포문인데요
 $J = \text{input} - k - 1$ 로 한 이유가
 일단 배열이니까 널값 포함되어있으니까 1 빼주
 고 6번째 페이지 포문에서 아랫줄 다 숫자 들어
 가게끔 해놔줄 거란 말이에요
 근데 이게 큰 포문 돌때마다 계속 반복될테니까
 k 로 한번 더 빼줬어요
 J 키워놓고 작게 한 이유가 이렇게 해야 숫자가
 아래에서 위로 커지니까 이렇게 해줬습니다

```
for (int i = 0; i < input; i++)  
{  
    for (int j = 0; j < input; j++)  
    {  
        printf(_Format: "%dWt", arr[i][j]);  
  
        if (int j == input - 1)  
        {  
            printf(_Format: "WnWn");  
        }  
    }  
}
```

출력해주는 포문이에요
l랑 j 키워가면서 출력해줬고
보기 좋으라고 Wt 해줬어요

달팽이 되게 하려고
J가 input - 1 될 때마다 WnWn
출력하게 했습니다


```
* ㅁ X 소스.c
파일 (전역 범위)
73 for (int i = 0; i < input; i++)
74 {
75     l = input - i - 1;
76     arr1 = arr[l][0];
77
78     for (int j = 1; j <= input - i - 1; j++)
79     {
80         arr1 *= arr[--l][j];
81     }
82     sum += arr1;
83 }
84 int arr2, o;
85 for (int i = 0; i < input - 1; i++)
86 {
87     o = input - 1;
88     arr2 = arr[o][i + 1];
89     for (int j = i + 1; j < input - 1; j++)
90     {
91         arr2 *= arr[--o][j + 1];
92     }
93     sum += arr2;
94 }
95
```

첫번째 포문이 배열 가운데줄부터 왼쪽 위에 대각선들 곱하고 더하는 포문이고 두번째 포문이 나머지 구하는 포문이에요

L 이 곱해지는 식?에서 항상 고정되어있어요 그래서 포문 안에서 l 1씩 작아지게 해놨으니 포문 끝나면 다시 초기화되게끔 해줬고

아래에 o도 똑같은 원리로 해줬어요 포문 안에 작은 포문들이 대각선 곱하는 식이고 sum에 다 더했습니다

```
}
printf(_Format: "대각선 곱 합: %d", sum);
return 0;
}
```

감사합니다