

2023-1 윈도우 프로그래밍 워밍업 문제 (3~6)

2023년도 1학기

3. [2차원 배열 사용하기] 4x5 행렬 만들고 행렬의 연산 구현하기

- 4행 5열의 행렬을 만들고 숫자(중복 불가)를 행렬에 저장한다.
 - 1 ~ 50 사이의 랜덤한 숫자를 행렬에 자동 입력
 - 열의 위치를 맞춰서 출력한다.
- 다음의 명령어를 수행한 후 결과를 출력한다. 모든 명령어는 다시 누르면 실행이 취소되고 원래의 값으로 변경된다.
 - a: 1행1열을 시작으로 **행을 따라 오름차순으로 정렬**
 - d: 1행5열을 시작으로 **열을 따라 내림차순으로 정렬**
 - e: 짝수만 출력하기
 - o: 홀수만 출력하기
 - m: 최대값 출력
 - n: 최소값 출력
 - p: 한 행의 값을 모두 더해 각 행의 1열에 저장
 - r: 기존의 숫자를 랜덤하게 재정렬
 - s: 새로운 랜덤한 숫자를 다시 자동 입력 (리셋)
 - q: 프로그램 종료

결과 예)

행렬값:

3	10	24	11	18
39	41	5	17	22
1	29	28	13	20
25	14	33	43	50

명령어 입력 d:

10	17	24	33	50
5	14	22	29	43
3	13	20	28	41
1	11	18	25	39

명령어 입력 m:

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	50

명령어 입력 p:

101	41	18	3	29
152	28	43	25	14
135	17	24	11	50
95	22	13	20	1

명령어 입력 a:

1	3	5	10	11
13	14	17	18	20
22	24	25	28	29
33	39	41	43	50

명령어 입력 e:

0	10	24	0	18
0	0	0	0	22
0	0	28	0	20
0	14	0	0	50

명령어 입력 r:

10	41	18	3	29
5	28	43	25	14
33	17	24	11	50
39	22	13	20	1

4. [구조체 사용하기] 텍스트를 사용하여 2인 돌 이동하기

- 화면에 10x10 바둑판을 그린다. 바둑판의 임의의 위치에 2개의 다른 돌을 놓고 사용자의 명령어(wasd 와 ijkl)에 따라 돌을 좌/우/상/하로 번갈아 입력 받아 움직인다.
 - 두 종류의 다른 돌은 번갈아 가며 움직일 수 있다.
 - 두 사용자의 이동 명령어는 두 종류의 다른 조합을 사용한다. (wasd와 ijkl)
- 가장자리에 돌이 닿으면 반대편 가장자리에 나타난다. 반대편에 나타날 때 돌의 모양이 바뀐다.
 - 돌의 모양은 5개 중 한 개로 랜덤하게 선택한다: 5개의 돌 모양 예) o, x, #, @, *
- 움직이는 돌이 다른 돌과 부딪치면
 - 움직인 돌 (위의 돌)의 모양만 나타난다. 이때 비프 소리를 출력한다.
 - 다음 순서인 아래의 돌을 움직이게 되면 바뀐 모양으로 나타난다.
- 돌의 위치와 상태를 가지고 있는 구조체를 사용한다.
- 키보드 명령으로 이동 키보드 외에 프로그램 리셋/종료 명령을 추가한다.
 - s: 프로그램 리셋 (시작위치에서 새로운 돌 모양으로 시작)
 - q: 프로그램 종료

** 비프음 소리내기: 함수 Beep 사용

```
#include <windows.h>
```

```
Beep (음의 높이, 음의 지속시간);
```

결과 예)

```
-----
| | |x| | | | | | |
-----
| | | | |O| | | | |
-----
...
```

Input command: j (x가 좌측 이동)

```
-----
| |x| | |O| | | | |
-----
| | | | | | | | | |
-----
...
```

Input command: a (O가 왼쪽으로 가니 x와 부딪치게 됨. O이 x위에 올려져서 O만 출력된다.)

```
-----
| | | |x|O| | | | |
-----
| | | | | | | | | |
-----
```

Input command: k (다음 순서로 x를 아래로 움직이면, x의 모양이 다른 모양으로 바뀌어 나타난다)

```
-----
| | | |O| | | | |
-----
| | | |@| | | | |
-----
```

Input command: w (원이 위로 이동)

```
-----
| | |x| |O| | | | |
-----
| | | | | | | | | |
-----
...
```

Input command: d (원이 우측으로 이동)

```
-----
| |x| | |O| | | | |
-----
| | | | | | | | | |
-----
...
```

...

.....

5. [switch-case문 사용하기] 달력 만들기

- 2023년 3월 1일은 수요일이다. 기본으로 2023년 3월의 달력을 출력한다.
- 사용자가 날짜를 입력하면
 - 입력하는 날짜의 요일을 계산하고,
 - 입력하는 달을 출력한다. (입력 날짜는 2000년부터 2030년까지로 제한, switch case 사용하기)
 - 키보드 f/F를 입력하면 2000년부터 2030년 사이의 해를 입력하면 그 해의 모든 달의 마지막 날의 날짜와 요일을 출력한다.
 - 윤년 계산을 한다.
 - 윤년 (2월이 29일까지인 해): 4년마다 윤년이 돌아오고, 100의 배수의 해는 윤년이 아니다. 그러나 400으로 나뉘는 해는 윤년이다.

결과 예)

2023/03/01: Wednesday

2023-3

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
			1	2	3	4

....

2026/12/31: Thursday

//--- 2026년 12월을 출력한다.

2023/06/21: Wednesday

//--- 2023년 6월을 출력한다.

f 2020	2020-01-31 Friday	2020-02-29 Saturday	2020-03-31 Tuesday	2020-04-30 Thursday
	2020-05-31 Sunday	2020-06-30 Tuesday	2020-07-31 Friday	2020-08-31 Tuesday
	2020-09-30 Wednesday	2020-10-31 Saturday	2020-11-30 Monday	2020-12-31 Thursday

6. [함수 사용하기] 계산 순위 정하여 정해진 순위 따라 연산하기

- 사용자가 입력하는 숫자 5개와 연산자 4개를 랜덤하게 설정한다.
- 숫자는 한자리 또는 두자리의 정수다.
- 사용자의 명령어를 입력받아 우선순위로 또는 순서대로 또는 역 순서로 계산하여 출력한다.
 - 명령어 1: 연산자 우선순위로
 - 명령어 2: 순서대로
 - 명령어 3: 역 순서대로
 - 명령어 4: 새로 숫자와 연산자 설정
 - 명령어 q: 프로그램 종료

결과 예)

Numbers to calculate: $3+4*5*2-1$

Input the calculating order: 1

Result: $3 + ((4*5)*2) - 1 = 42$ //--- 연산자의 우선순위로 계산한다.

Input the calculating order: 2

Result: $((((3+4)*5)*2)-1 = 69$ //--- 앞에서부터 순서대로 계산한다.

Input the calculating order: 3

Result: $3+(4*(5*(2-1))) = 23$ //--- 뒤에서부터 계산한다.

Input the calculating order: 4

Numbers to calculate: $3+6/4*2+5$

Input the calculating order: 1

Result: $3 + ((6/4) *2) + 5 = 11$

...

워밍업 프로그램은

- 문제 당 채점 기준
 - 1문제당 3점으로 계산하여 적용됨
 - 각 문제당 100% 이상 구현했을 때: 3점
 - 각 문제당 50% 이상 구현했을 때: 2점
 - 각 문제당 30% 이상 구현했을 때: 1점
- 본인의 코딩 워밍업이니 최대한 스스로의 힘으로 구현하세요!