

### C프로그래밍 과제 #3 (라이브러리 함수 연습)

□ 문제 : 프로그램이 시작되면 임의의 위치(x좌표는 10~70 사이값, y좌표는 5~20 사이값)에 '\*' 문자 하나가 나타난다. 그리고 '\*' 문자는 0.5초 간격으로 오른쪽으로 이동한다. 즉, x 좌표의 값이 1씩 증가하는 것이다. 그런데 방향키 중 하나를 누르면 해당 방향으로 이동 방향이 변경되며 그 다음부터는 그 방향으로 한 칸씩 이동하게 된다. 만약 '\*' 문자가 벽에 부딪히면 반대 방향으로 방향을 바꾸어 이동하게 된다. ESC 키를 입력하면 프로그램은 종료한다.

- 수업 자료 “주요 라이브러리 함수들.pptx” 파일의 종합 예제를 참고하여 만들면 된다. 해당 함수들을 그대로 사용할 수 있다.

□ 과제 보고서 작성 방법 제출 방법

- 워드프로세서(한글)로 작성
- 프로그램 소스 파일, 실행 결과(예) 포함
- 프로그램에 문제가 있는 경우 문제점 기술
- 제출 방법 : 강의지원시스템 해당 과제에 파일 업로드 (파일명 : 학번\_이름.hwp)
- 보고서 파일(한글)의 코드를 Visual Studio로 복사하면 실행될 수 있도록 만들(확인 바람)

□ 코드

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
#include <conio.h>
#include <Windows.h>

#define KEY_LEFT (256+75)
#define KEY_UP (256+72)
#define KEY_RIGHT (256+77)
#define KEY_DOWN (256+80)
#define KEY_ESC 27

int GetKey(void);
double GetElapsedTime(clock_t initial_clock, clock_t current_clock);
void GotoXY(int x, int y);
void Erase(int x, int y);    // (x, y)로 이동하여 공백 문자 출력
void Draw(int x, int y);    // (x, y)로 이동하여 '*' 문자 출력

int main(void)
{
    system("mode con cols=70 lines=20");
    srand(time(NULL));
```

```
int x = rand() % 61 + 10;
int y = rand() % 16 + 5;
int Direction = 1; // 1 = RIGHT, 2 = LEFT, 3 = UP, 4 = DOWN
Draw(x, y);
```

```
clock_t initial_clock = clock();
```

```
while (1)
```

```
{
```

```
    clock_t current_clock = clock();
```

```
    if (x == 0)
```

```
        Direction = 1;
```

```
    if (x == 69)
```

```
        Direction = 2;
```

```
    if (y == 0)
```

```
        Direction = 4;
```

```
    if (y == 19)
```

```
        Direction = 3;
```

```
    if (GetElapsedTime(initial_clock, current_clock) > 0.5)
```

```
    { // 0.5초 경과
```

```
        switch(Direction)
```

```
        {
```

```
        case 1 :
```

```
            Erase(x, y);
```

```
            x++;
```

```
            Draw(x, y);
```

```
            break;
```

```
        case 2 :
```

```
            Erase(x, y);
```

```
            x--;
```

```
            Draw(x, y);
```

```
            break;
```

```
        case 3 :
```

```
            Erase(x, y);
```

```
            y--;
```

```
            Draw(x, y);
```

```

        break;

    case 4 :
        Erase(x, y);
        y++;
        Draw(x, y);
        break;
    }
    initial_clock = current_clock; // 기준 시각 재설정
}
if (_kbhit())
{
    int key = GetKey();

    if (key == KEY_RIGHT)
    {
        Direction = 1;
    }
    else if (key == KEY_LEFT)
    {
        Direction = 2;
    }
    else if (key == KEY_UP)
    {
        Direction = 3;
    }
    else if (key == KEY_DOWN)
    {
        Direction = 4;
    }
    else if (key == KEY_ESC)
        break;
}
}

int GetKey(void)
{
    int ch = _getch();

    if (ch == 0 || ch == 224)
        // 방향키의 경우 0 또는 224의 값이 먼저 입력됨

```

```

        ch = 256 + _getch();
        // 그 다음에 해당 방향키에 따라 72(Up),
        // 80(Down), 75(Left), 77(Right) 값이 입력됨
        return ch;
    }

double GetElapsedTime(clock_t initial_clock, clock_t current_clock)
{
    return (double)(current_clock - initial_clock) / CLOCKS_PER_SEC;
}

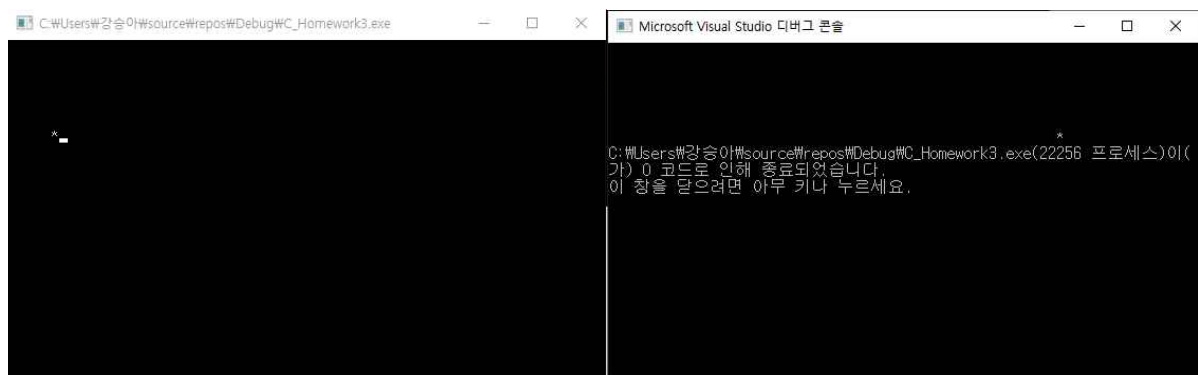
void GotoXY(int x, int y)
{
    // COORD 구조체 변수를 통해 이동할 위치 설정
    COORD pos = { x, y };
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), pos);
}

void Erase(int x, int y)
{
    GotoXY(x, y);
    printf(" ");
}

void Draw(int x, int y)
{
    GotoXY(x, y);
    printf("*");
}

```

#### □ 실행 화면



< 방향키에 따라 방향 전환하며 벽을 만나면 반대방향으로 진행, esc를 누르게 되면 종료된다. >