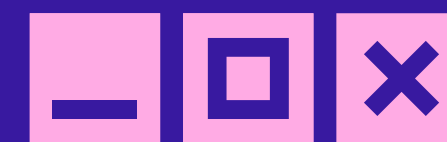
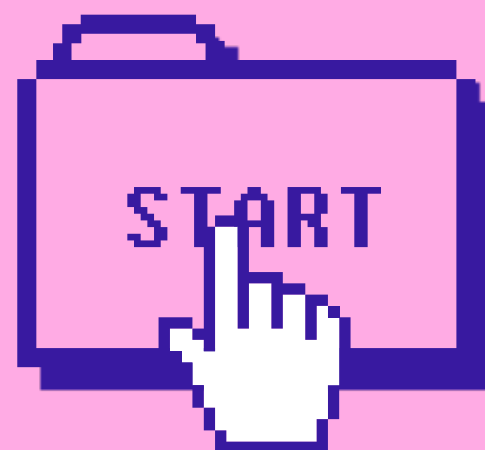


사용언어 : C



지령이 게임

[2022 SWING 내부 프로젝트]



PRESENTATION

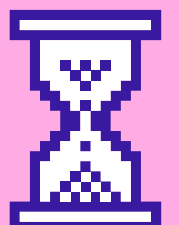
OVERVIEW

함수 설명

어려웠던 점

향후 발전가능성

프로젝트로 얻은 것





```
void filed() { // 게임 영역
```

```
    for (int i = 1; i < FIELD_WIDTH; i++)  
    {  
        gotoXY(i, 0, "-");  
        gotoXY(i, FIELD_HEIGHT, "-");  
    }
```

```
    for (int i = 1; i < FIELD_HEIGHT; i++)  
    {  
        gotoXY(0, i, "|");  
        gotoXY(FIELD_WIDTH, i, "|");  
    }
```

```
    gotoXY(0, 0, "┌");
```

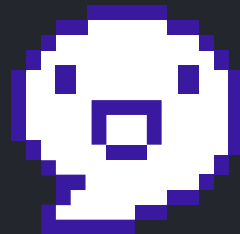
```
    gotoXY(0, FIELD_HEIGHT, "└");
```

```
    gotoXY(FIELD_WIDTH, 0, "┐");
```

```
    gotoXY(FIELD_WIDTH, FIELD_HEIGHT, "┘");
```

```
    gotoXY(FIELD_WIDTH, 30, "\n");
```

```
}
```



게임 영역 함수

게임 영역을 출력하는 함수를 구현

#define으로 영역의 값을 고정 후 반복문으로 코드를 간략화

gotoXY 함수로 좌표값을 지정해 출력

```
void title() {  
  
    filed();  
  
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 5, "

S N A K E

");  
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 6, "

S N A K E

");  
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 7, "

S N A K E

");  
  
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 9, " [ PRESS ANY KEY TO START ] ");  
  
    while (1) {  
        if (_kbhit()) { //키입력받음  
            key = _getch();  
            if (key == 27) {  
                exit(0); // ESC키면 종료  
            }  
            else {  
                break; //아니면 그냥 계속 진행  
            }  
        }  
        gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 9, " [ PRESS ANY KEY TO START ] ");  
        Sleep(400);  
        gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 9, "

S N A K E

");  
        Sleep(400);  
    }  
  
    reset();  
}
```



게임 타이틀 함수

게임 타이틀 화면을 출력하고 동작하는 함수 구현
반복문을 사용해 키를 입력 받고 아무 키나 누르면
게임 시작 ESC 키 누르면 게임 종료



```
void timer() {  
    for (int sec = 10; sec > -1; sec--)  
    {  
        gotoXY(MAP_X - 2, MAP_Y + FIELD_HEIGHT, " TIME: ");  
        printf("%d초..", sec);  
        Sleep(1000);  
  
        if (sec == 0)  
        {  
            gameOver();  
        }  
    }  
}
```



게임 타이머 함수

게임 내의 타이머 함수를 구현
반복문과 sleep 함수를 사용하여 지정한 시간이
될 시 종료 함수 호출



```
void reset() {  
    system("cls");  
  
    filed();  
  
    while (_kbhit()) _getch();  
  
    prey_m = 0;  
    score = 0;  
  
    prey();  
    timer();  
}
```



게임 초기화 함수

게임 내의 초기화 함수를 구현

시스템 명령어 cls를 사용해 콘솔창에 출력 된 것을 지우고,
새로 게임 영역 출력 후 와일문을 사용해 그 동안 저장했던

키의 값을 삭제

그 외의 변수도 초기값으로 설정

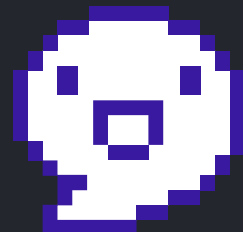
```
void gameOver() {
    system("cls");
    filed();

    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 5, "
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 6, "
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 7, "

    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 10, MAP_Y + 9, "YOUR SCORE : ");
    printf(" %d ! ", score);

    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 13, "  ◇ SPACE BAR : Restart");
    gotoXY(MAP_X + (FIELD_WIDTH / 2) - 15, MAP_Y + 14, "  ◇ ESC : Quit");

    while (1)
    {
        if (_kbhit()) { //키입력받음
            key = _getch();
            if (key == 32) { // spacebar 누르면 재시작
                reset();
            }
            else if (key == 27) {
                exit(1); //ESC 키 누르면 종료
            }
        }
    }
}
```



게임 종료 함수

게임 종료 함수 구현

게임 진행한 콘솔 삭제 후 다시 게임 영역 출력
반복문으로 구현하였기 때문에,
사용자가 종료하기 전까지 게임 플레이 가능



```
void prey() {  
  
    gotoXY(MAP_X + 8, MAP_Y + FIELD_HEIGHT, " "); //점수표시  
    printf("SCORE: %3d", score);  
  
    while (1) {  
  
        if (prey_m == 0) { //prey_m의 초기값이 0이므로 처음에는 무조건 출력된다  
            //먹이 랜덤 x, y좌표를 food_x, food_y에 각각 저장.  
            prey_x = preyX();  
            prey_y = preyY();  
  
            gotoXY(prex_x, prey_y, "♣"); // x, y좌표에 먹이를 뿌려줌  
            prey_m++;  
        }  
        else if (prey_m == 1)  
        {  
            for (int prey = 0; prey < prey_m; prey++)  
            {  
                prey_x = preyX();  
                prey_y = preyY();  
  
                gotoXY(prex_x, prey_y, "♣"); // x, y좌표에 먹이를 뿌려줌  
            }  
            prey_m++;  
        }  
        else {  
            printf("먹이는 최대 2개까지 생성됩니다.");  
        }  
        break;  
    }  
}
```



지렁이 먹이 함수

지렁이의 먹이를 콘솔창에 출력하는 함수 구현
콘솔창의 총 2개의 먹이만 출력하고자 if, for 문 사용
좌표 값을 랜덤으로 받아와 먹이 출력
실행 할 때마다 출력의 위치 변동


```
void noCursor() { // 콘솔창의 커서 숨김  
    HANDLE console;  
    CONSOLE_CURSOR_INFO ConsoleCursor;  
  
    console = GetStdHandle(STD_OUTPUT_HANDLE);  
    ConsoleCursor.bVisible = 0;  
}
```



콘솔창 커서 삭제 함수

**핸들러를 사용해 콘솔창의 커서를 제어할 수 있지만
왜 인지 모르게 적용 안됨.
오류가 나지 않아서 여러차례 코드를 수정했지만
해결 불가..**



```
void move() {
    int i;
    gameStatus = 1; //게임 실행

    earthWorm_x = 10;
    earthWorm_y = 10;

    if (GetAsyncKeyState(VK_LEFT) & 0x8000) { //왼쪽
        earthWorm_x--;
    }
    if (GetAsyncKeyState(VK_RIGHT) & 0x8000) { //오른쪽
        earthWorm_x++;
    }
    if (GetAsyncKeyState(VK_UP) & 0x8000) { //위
        earthWorm_y--;
    }
    if (GetAsyncKeyState(VK_DOWN) & 0x8000) { //아래
        earthWorm_y++;
    }

    gotoXY(earthWorm_x, earthWorm_y, "●");

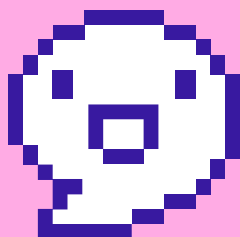
    if (earthWorm_x == prey_x && earthWorm_y == prey_y) { //먹이 먹은 경우
        score += 10;
        gotoXY(earthWorm_x - 1, earthWorm_y - 1, "○");
    }

    if (earthWorm_x == FIELD_WIDTH - 1 || earthWorm_y == FIELD_HEIGHT - 1) {
        gameOver();
        return;
    }

    for (i = earth_len - 1; i > 0; i--) { //몸통좌표를 한칸씩 옮김
        earthWorm_x = x[i - 1];
        earthWorm_y = y[i - 1];
    }
}
```

지렁이 이동 함수

오류를 고치지 못하기도 하였고
getch() 함수는 대각선 이동이 불가능한 관계로
중간 발표 때와는 완전히 다른 함수를 사용



GetAsyncKeyState()

if(GetAsyncKeyState(VK_LEFT) & 0x8000)

반환값	설명
0(0x0000)	이전에 누른 적이 없고 호출 시점에서 안눌린 상태
0x8000	이전에 누른 적이 없고 호출 시점에서 눌린 상태
0x8001	이전에 누른 적이 있고 호출 시점에서 눌린 상태
1(0x0001)	이전에 누른 적이 있고 호출 시점에서 안눌린 상태

오류 목록

전체 솔루션

✖ 4 오류

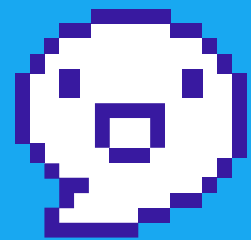
⚠ 0 경고

ℹ 0/1 메시지

	코드	설명
abc	E0008	닫는 따옴표가 없습니다.
✖	C2001	상수에 줄 바꿈 문자가 있습니다.
✖	C2137	문자 상수가 비어 있습니다.
✖	C2059	구문 오류: '상수'

어려웠던 점

1. 지렁이가 길어지게 하고 싶었으나 구현하지 못했다.
2. 함수 각각은 작동하는데 함수들을 합치고 빌드하니 오류가 뜨는데 고칠 수가 없었다.
3. 디버깅하지 않고 실행해도 먹이와 지렁이가 화면에 동시에 나타나게 할 수 없었다.



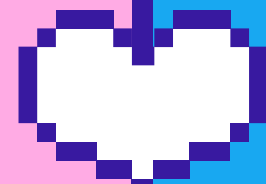
오류 수정

지렁이의 길이를 길어지게 하고 방향 전환 모션을 구현
먹이를 먹을 때마다 길어지게 하기
먹이 대신 다른 지렁이를 잡아먹되
큰 지렁이를 만나면 GAME OVER

향후 발전 가능성



프로젝트를
하면서...



C언어 숙련도 향상

팀원들 간의 협력

프로그래밍에 대한 이해도 및 알고리즘 구현 능력 향상

지금까지 흐물한 지렁이팀의 발표였습니다!

감사합니다.
발표에 대한 질의응답은 언제나 환영입니다!

30기 박지원, 손효림

