

Assignment 3 : ให้นักศึกษาเพิ่มเติมโปรแกรมให้มีการแสดงผลคะแนนที่มูมบนขวาของจอภาพ โดยคะแนนจะเพิ่มขึ้น เมื่อมีการยิงกระสุนไปทำลายดาว และจะมีการสุ่มตำแหน่งการเกิดดาวใหม่ทุกครั้งที่ถูกทำลาย พร้อมทั้งมีเสียง Beep ถ้ามีการยิงกระสุนโดนดาว

การส่งงาน : ให้นักศึกษาส่งโปรแกรมที่สมบูรณ์ของ Assignment 3 เพียงโปรแกรมเดียวเท่านั้น

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

void gotoxy(int x, int y)
{
    COORD c = { x, y };
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), c);
}
void setcolor(int fg, int bg)
{
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    SetConsoleTextAttribute(hConsole, bg * 16 + fg);
}
void draw_ship(int x, int y)
{
    gotoxy(x, y);
    setcolor(2, 7);
    printf("-<0>-");
}
void erase_ship(int x, int y)
{
    gotoxy(x, y);
    setcolor(7, 0);
    printf(" ");
}
void setcursor(bool visible)
{
    HANDLE console = GetStdHandle(STD_OUTPUT_HANDLE);
    CONSOLE_CURSOR_INFO lpCursor;
    lpCursor.bVisible = visible;
    lpCursor.dwSize = 20;
    SetConsoleCursorInfo(console, &lpCursor);
}
void draw_bullet(int x, int y)
{
    setcolor(7, 0);
    gotoxy(x, y);
    printf("^");
}
void clear_bullet(int x, int y)
{
    setcolor(7, 0);
    gotoxy(x, y);
    printf(" ");
}
int randrange(int start, int stop)
{

```

```

    return(rand() % (stop - start + 1)) + start;
}
void draw_star()
{
    for (int i = 0; i < 20; i++)
    {
        setcolor(7, 0);
        gotoxy(randrange(10, 70), randrange(2, 5));
        printf("*");
    }
}
char cursor(int x, int y) {
    HANDLE hStd = GetStdHandle(STD_OUTPUT_HANDLE);
    char buf[2]; COORD c = { x,y }; DWORD num_read;
    if (!ReadConsoleOutputCharacter(hStd, (LPTSTR)buf, 1, c, (LPDWORD)&num_read))
        return '\\0';
    else
        return buf[0];
}
int main()
{
    int x = 38, y = 28, direct = 0, bullet[5] = { 0,0,0,0,0 }, bx[5], by[5],
        bulletnow = 0, score = 0;
    setcolor(2, 4);
    srand(time(NULL));
    draw_star();
    char ch = '.';
    gotoxy(x, y);
    draw_ship(x, y);
    do {
        setcursor(0);
        if (_kbhit()) {
            ch = _getch();
            if (ch == 'a') {
                direct = 1;
            }
            if (ch == 'd')
            {
                direct = -1;
            }
            if (ch == 's')
            {
                direct = 0;
            }
            if (ch == ' ' && bullet[bulletnow] == 0)
            {
                bx[bulletnow] = x + 3, by[bulletnow] = y - 1;
                bullet[bulletnow] = 1;
                bulletnow++;
                bulletnow %= 5;
            }
            fflush(stdin);
        }
        if (direct == 1 && x > 0)
        {
            erase_ship(x, y);
            draw_ship(--x, y);
        }
        if (direct == -1 && x < 80)
        {
            erase_ship(x, y);
            draw_ship(++x, y);
        }
    } while (1);
}

```

```

    }
    for (int i = 0; i < 5; i++)
        if (bullet[i] == 1)
        {
            clear_bullet(bx[i], by[i]);
            if (by[i] > 0)
            {
                draw_bullet(bx[i], --by[i]);
                if (cursor(bx[i], by[i] - 1) == '*')
                {
                    score++;
                    gotoxy(90, 0);
                    printf("score=%d", score);
                    gotoxy(bx[i], by[i] - 1);
                    printf(" ");
                    Beep(500, 100);
                    clear_bullet(bx[i], by[i]);
                    bullet[i] = 0;
                    gotoxy(randrange(10, 70), randrange(1, 6));
                    printf("*");
                }
            }
            else bullet[i] = 0;
        }
    Sleep(100);
} while (ch != 'x');
return 0;
}

```