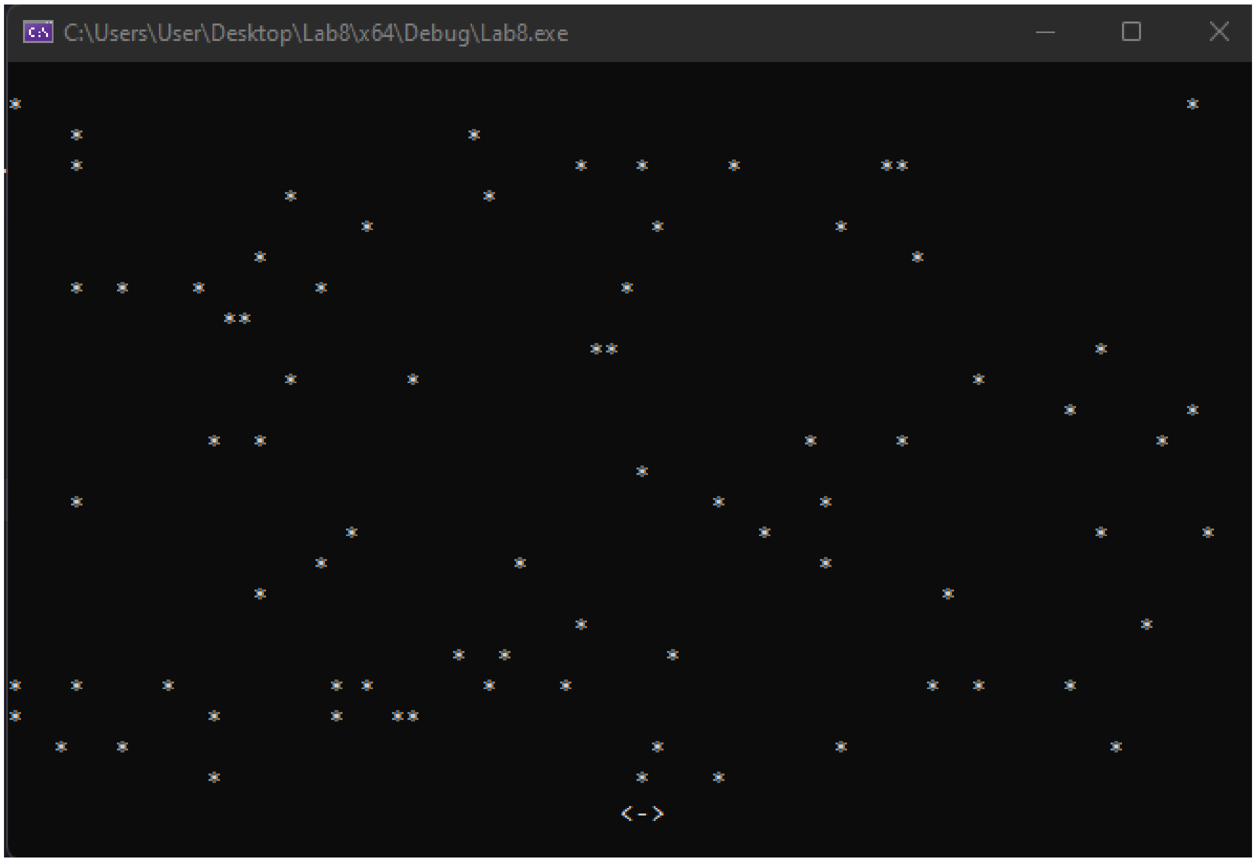
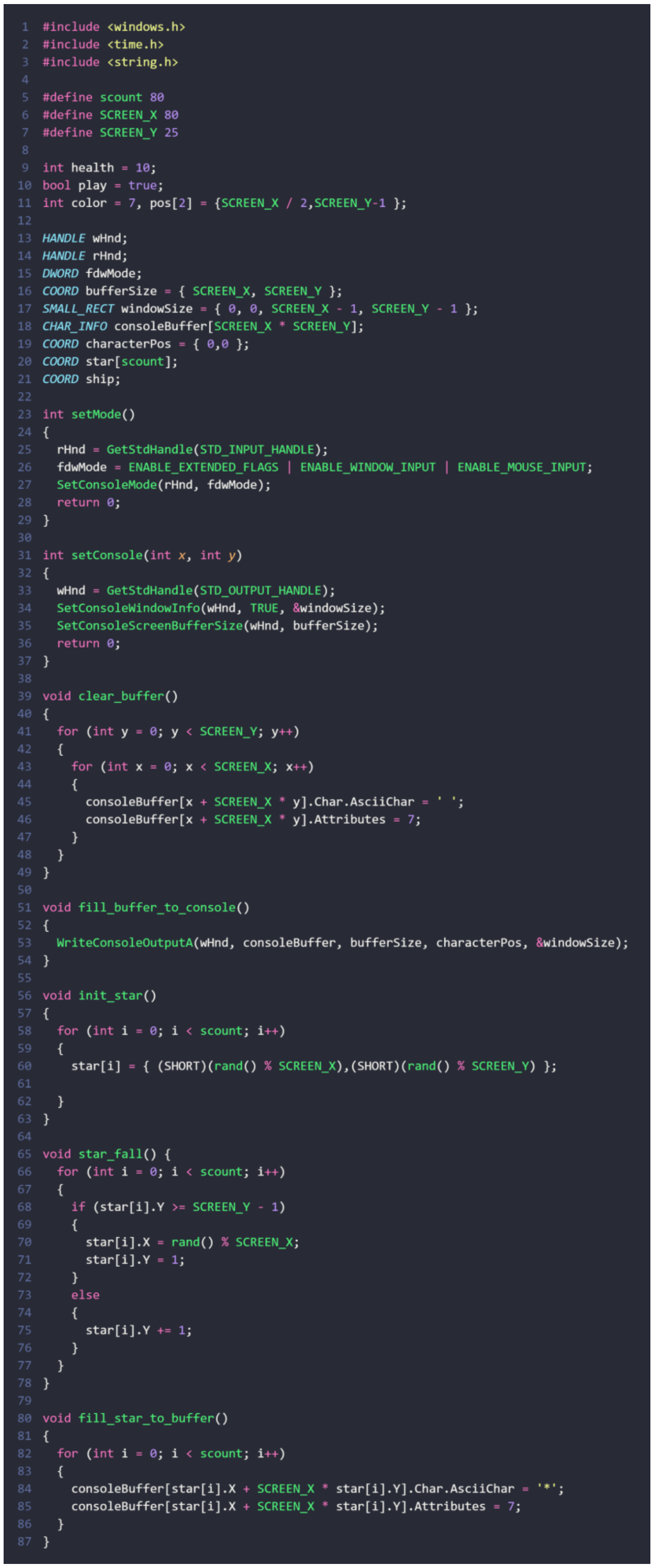
65010077 - นายกิตติพศ หลำบางช้าง



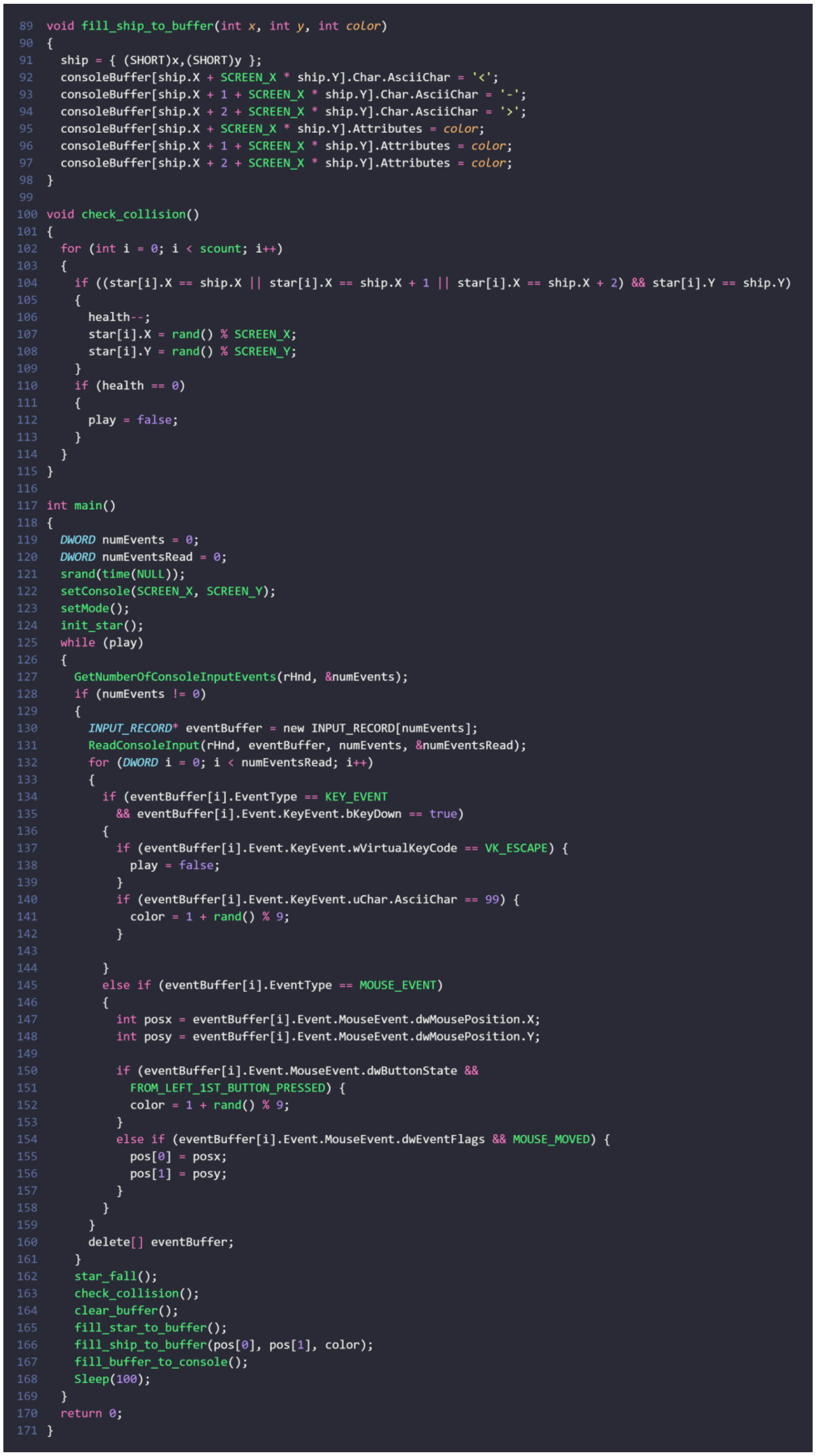
C:\Users\User\Desktop\Lab8\x64\Debug\Lab8.exe

```
*                                                                            *
    *                         *
    *                                *   *     *         **
                  *            *
                       *                  *           *
                *                                          *
    *  *    *       *                   *
              **
                                      **                               *
                  *       *                                    *
                                                                     *       *
             *  *                                   *     *                *
                                         *
    *                                         *      *
                      *                          *                     *      *
                    *            *                   *
                *                                            *
                                     *                                    *
                             *  *          *
*   *     *          * *       *    *                       *  *     *
*            *       *   **
   *  *                                  *           *                 *
             *                           *    *
                                        <->
```

65010077 - นายกิตติพศ หลำบางช้าง



```c
#include <windows.h>
#include <time.h>
#include <string.h>

#define scount 80
#define SCREEN_X 80
#define SCREEN_Y 25

int health = 10;
bool play = true;
int color = 7, pos[2] = {SCREEN_X / 2,SCREEN_Y-1 };

HANDLE wHnd;
HANDLE rHnd;
DWORD fdwMode;
COORD bufferSize = { SCREEN_X, SCREEN_Y };
SMALL_RECT windowSize = { 0, 0, SCREEN_X - 1, SCREEN_Y - 1 };
CHAR_INFO consoleBuffer[SCREEN_X * SCREEN_Y];
COORD characterPos = { 0,0 };
COORD star[scount];
COORD ship;

int setMode()
{
  rHnd = GetStdHandle(STD_INPUT_HANDLE);
  fdwMode = ENABLE_EXTENDED_FLAGS | ENABLE_WINDOW_INPUT | ENABLE_MOUSE_INPUT;
  SetConsoleMode(rHnd, fdwMode);
  return 0;
}

int setConsole(int x, int y)
{
  wHnd = GetStdHandle(STD_OUTPUT_HANDLE);
  SetConsoleWindowInfo(wHnd, TRUE, &windowSize);
  SetConsoleScreenBufferSize(wHnd, bufferSize);
  return 0;
}

void clear_buffer()
{
  for (int y = 0; y < SCREEN_Y; y++)
  {
    for (int x = 0; x < SCREEN_X; x++)
    {
      consoleBuffer[x + SCREEN_X * y].Char.AsciiChar = ' ';
      consoleBuffer[x + SCREEN_X * y].Attributes = 7;
    }
  }
}

void fill_buffer_to_console()
{
  WriteConsoleOutputA(wHnd, consoleBuffer, bufferSize, characterPos, &windowSize);
}

void init_star()
{
  for (int i = 0; i < scount; i++)
  {
    star[i] = { (SHORT)(rand() % SCREEN_X),(SHORT)(rand() % SCREEN_Y) };

  }
}

void star_fall() {
  for (int i = 0; i < scount; i++)
  {
    if (star[i].Y >= SCREEN_Y - 1)
    {
      star[i].X = rand() % SCREEN_X;
      star[i].Y = 1;
    }
    else
    {
      star[i].Y += 1;
    }
  }
}

void fill_star_to_buffer()
{
  for (int i = 0; i < scount; i++)
  {
    consoleBuffer[star[i].X + SCREEN_X * star[i].Y].Char.AsciiChar = '*';
    consoleBuffer[star[i].X + SCREEN_X * star[i].Y].Attributes = 7;
  }
}
```



```cpp
void fill_ship_to_buffer(int x, int y, int color)
{
  ship = { (SHORT)x,(SHORT)y };
  consoleBuffer[ship.X + SCREEN_X * ship.Y].Char.AsciiChar = '<';
  consoleBuffer[ship.X + 1 + SCREEN_X * ship.Y].Char.AsciiChar = '-';
  consoleBuffer[ship.X + 2 + SCREEN_X * ship.Y].Char.AsciiChar = '>';
  consoleBuffer[ship.X + SCREEN_X * ship.Y].Attributes = color;
  consoleBuffer[ship.X + 1 + SCREEN_X * ship.Y].Attributes = color;
  consoleBuffer[ship.X + 2 + SCREEN_X * ship.Y].Attributes = color;
}

void check_collision()
{
  for (int i = 0; i < scount; i++)
  {
    if ((star[i].X == ship.X || star[i].X == ship.X + 1 || star[i].X == ship.X + 2) && star[i].Y == ship.Y)
    {
      health--;
      star[i].X = rand() % SCREEN_X;
      star[i].Y = rand() % SCREEN_Y;
    }
    if (health == 0)
    {
      play = false;
    }
  }
}

int main()
{
  DWORD numEvents = 0;
  DWORD numEventsRead = 0;
  srand(time(NULL));
  setConsole(SCREEN_X, SCREEN_Y);
  setMode();
  init_star();
  while (play)
  {
    GetNumberOfConsoleInputEvents(rHnd, &numEvents);
    if (numEvents != 0)
    {
      INPUT_RECORD* eventBuffer = new INPUT_RECORD[numEvents];
      ReadConsoleInput(rHnd, eventBuffer, numEvents, &numEventsRead);
      for (DWORD i = 0; i < numEventsRead; i++)
      {
        if (eventBuffer[i].EventType == KEY_EVENT
          && eventBuffer[i].Event.KeyEvent.bKeyDown == true)
        {
          if (eventBuffer[i].Event.KeyEvent.wVirtualKeyCode == VK_ESCAPE) {
            play = false;
          }
          if (eventBuffer[i].Event.KeyEvent.uChar.AsciiChar == 99) {
            color = 1 + rand() % 9;
          }

        }
        else if (eventBuffer[i].EventType == MOUSE_EVENT)
        {
          int posx = eventBuffer[i].Event.MouseEvent.dwMousePosition.X;
          int posy = eventBuffer[i].Event.MouseEvent.dwMousePosition.Y;

          if (eventBuffer[i].Event.MouseEvent.dwButtonState &&
            FROM_LEFT_1ST_BUTTON_PRESSED) {
            color = 1 + rand() % 9;
          }
          else if (eventBuffer[i].Event.MouseEvent.dwEventFlags && MOUSE_MOVED) {
            pos[0] = posx;
            pos[1] = posy;
          }
        }
      }
      delete[] eventBuffer;
    }
    star_fall();
    check_collision();
    clear_buffer();
    fill_star_to_buffer();
    fill_ship_to_buffer(pos[0], pos[1], color);
    fill_buffer_to_console();
    Sleep(100);
  }
  return 0;
}
```