

**Національний технічний університет України  
“Київський політехнічний інститут  
імені Ігоря Сікорського”**

Факультет прикладної математики  
Кафедра системного програмування і спеціалізованих  
комп'ютерних систем

***Розрахункова робота***

з дисципліни

**« Програмування 2. Складні структури »**

Виконав : Майстренко Єгор

Група : КВ — 04

Варіант : 13

Перевірила : Тарасенко-Клятченко

Київ — 2021

## **Завдання**

1. Створити ігрову програму мовою програмування C.
2. Розробка і реалізація ігрових програм має вестися з врахуванням графічних та звукових можливостей, що надаються конкретним комп'ютером.
3. Програма мусить коректно розв'язувати поставлену задачу. Логічно відокремлені частини алгоритму реалізувати за допомогою окремих функцій.
4. Також потрібно передбачити та забезпечити виконання всіх можливих розгалужень алгоритму, тобто програма повинна коректно реагувати на будь-які можливі ситуації (наприклад, виникнення помилкових ситуацій, перевірка файлів на порожність, правильність введених з клавіатури значень і т. д.). Передбачити взаємодію з користувачем (наприклад, можливість виводу правил гри, допомоги), таймер, лічильник числа ходів відповідно до поставленої в конкретному варіанті задачі.

### **Варіант 13**

Потрібно ввести курсор в позначену область екрана (невеликий кружечок), розташування якого невідоме гравцю. Рух курсора супроводжується звуковим сигналом: якщо курсор наближається до області, то звук стає вищим, якщо віддаляється – то нижчим.

## **Опис структури програми**

void Menu() - функція з виводу інформації для вибору користувачем різних варіантів продовження взаємодії.

void Button() - функція, що виводить повідомлення про неправильно вибрану клавішу користувачем для подальшого переходу по меню.

void Info() - функція з правилами гри та корисною інформацією для гравця.

void Zastavka() - головна функція, що викликає заставку з інформацією про творця.

void Game() - функція, що викликає спливаюче вікно, яке й дає змогу грати.

void End\_Game() - функція для виводу інформації та вибору користувачем наступних дій після перемоги у грі.

void Generation() - функція для генерації випадкової позиції кола, яке необхідно знайти гравцю.

void WinShow() - функція, що створює зону, яку необхідно знайти гравцю. Також фарбує цю зону у вибраний колір.

LRESULT WndProc — функція, що генерує звукові сигнали, визначає чи переміг користувач та рахує відстань між вибраною частиною вікна та розташуванням шуканого кола.

## Текст програми

```
#include <stdlib.h>
#include <stdio.h>
#include <windows.h>
#include <math.h>
#include <time.h>
#include <conio.h>
//-----
// Ініціалізація змінних для збереження певних даних
RECT rct;
int Round_X;
int Round_Y;
//-----
void Menu();
void Button();
void Info();
void Zastavka();
void Game();
void End_Game();
void Generation();
//-----
void End_Game() // інтерфейс меню - завершення гри
{
    system("cls");
    printf("\n\n\n\n");
    printf("\t\t\t\t\t ----- \n");
    printf("\t\t\t\t\t THE END! YOU ARE SUCH A CLEVER BOY | \n");
    printf("\t\t\t\t\t ----- \n");
    printf("\n\n");
    printf("\n\t\t\t\t\t Press 1 to restart the game\n");
    printf("\n\t\t\t\t\t Press 2 to go to the menu\n");
    printf("\n\t\t\t\t\t Press 3 to quit the program\n");
    switch(getch())
    {
        case '1':
            system("cls");
            Game();
            break;
        case '2':
            system("cls");
            Menu();
            break;
        case '3':
            system("cls");
            exit(0);
        default :
            Button();
            End_Game();
    }
}
//-----
void Button() // інтерфейс меню для виводу повідомлення про неправильно
натиснуту клавішу
{
    system("cls");
    printf("\n\n\n\n\n\n\n\n\n\n");
    printf("\t\t\t\t\t ----- \n");
    printf("\t\t\t\t\t CHOOSE THE CORRECT BUTTON !!! | \n");
    printf("\t\t\t\t\t ----- \n");
    Sleep(750);
}
```

```
//-----  
void Info() // інтерфейс меню для виводу правил гри  
{  
    system("cls");  
    printf("\n\n\n\n");  
    printf("\t\t\t\t\t-----\n");  
    printf("\t\t\t\t\tHi! In this game you need to \n");  
    printf("\t\t\t\t\tlook for hidden circle in the \n");  
    printf("\t\t\t\t\twindow with help of the sound. \n");  
    printf("\t\t\t\t\tWhen the game start, use your \n");  
    printf("\t\t\t\t\tmouse to click on Sound Game \n");  
    printf("\t\t\t\t\twindow. Be calm, because if you \n");  
    printf("\t\t\t\t\tclick too much, program will \n");  
    printf("\t\t\t\t\treply with a sum of sounds you \n");  
    printf("\t\t\t\t\tclicked. Don't be scare of it! \n");  
    printf("\t\t\t\t\tYou should click every 1,5 seconds \n");  
    printf("\t\t\t\t\tIf you reach the hide circle you \n");  
    printf("\t\t\t\t\twill win the game. For more \n");  
    printf("\t\t\t\t\tinformation you can contact \n");  
    printf("\t\t\t\t\twrite a creator to his mail \n");  
    printf("\t\t\t\t\tGood luck! \n");  
    printf("\t\t\t\t\t-----\n");  
    printf("\n\n");  
    printf("\n\t\t\t\t\tPress 1 to go to the menu\n");  
    printf("\n\t\t\t\t\tPress 2 to quit the program\n");  
    switch(getch())  
    {  
        case '1':  
            system("cls");  
            Menu();  
            break;  
        case '2':  
            system("cls");  
            exit(0);  
        default :  
            Button();  
            Info();  
    }  
}  
//-----  
void Zastavka() // інтерфейс меню - інформація стосовно роботи  
{  
    system("cls");  
    printf("\n\n\n\n");  
    printf("\t\t\t\t\t-----\n");  
    printf("\t\t\t\t\t\n");  
    printf("\t\t\t\t\t_ GAME MAKING _ \n");  
    printf("\t\t\t\t\tVariant: 13 \n");  
    printf("\t\t\t\t\tMade by Yegor Maystrenko \n");  
    printf("\t\t\t\t\tgroup KV - 04 \n");  
    printf("\t\t\t\t\t-----\n");  
    printf("\n\n");  
    printf("\n\t\t\t\t\tPress 1 to start the program\n");  
    printf("\n\t\t\t\t\tPress 2 to quit the program\n");  
    switch(getch())  
    {  
        case '1':  
            system("cls");  
            Menu();  
            break;  
        case '2':
```

```

        system("cls");
        exit(0);
    default :
        Button();
        Zastavka();
    }
}

//-----
void Menu() // інтерфейс меню - головне меню
{
    system("cls");
    printf("\n\n\n\n");
    printf("\t\t\t\t\t-----\n");
    printf("\t\t\t\t\t|                               |\n");
    printf("\t\t\t\t\t|           _ MENU _           |\n");
    printf("\t\t\t\t\t|                               |\n");
    printf("\t\t\t\t\t|           Press 1 to play the game           |\n");
    printf("\t\t\t\t\t|                               |\n");
    printf("\t\t\t\t\t|           Press 2 to open game rules           |\n");
    printf("\t\t\t\t\t|                               |\n");
    printf("\t\t\t\t\t|           Press 3 to go back           |\n");
    printf("\t\t\t\t\t|                               |\n");
    printf("\t\t\t\t\t|           Press 4 to quit completely           |\n");
    printf("\t\t\t\t\t-----\n");
    switch(getch())
    {
        case '1':
            system("cls");
            Game();
            break;
        case '2':
            system("cls");
            Info();
            break;
        case '3':
            system("cls");
            Zastavka();
            break;
        case '4':
            system("cls");
            exit(0);
        default :
            Button();
            Menu();
    }
}

//-----
void Generation()
{
    srand(time(NULL));
    Round_X = rand() % 650; // Генерація центру кола для осі X
    Round_Y = rand() % 650; // Генерація центру кола для осі Y
}

//-----
void WinShow(HDC dc) // Функція, яка створює зону для звукового сигналу для пізнішої взаємодії з консоллю
{
    HDC memDC = CreateCompatibleDC(dc);
    HBITMAP memBM = CreateCompatibleBitmap(dc, rct.right - rct.left, rct.bottom - rct.top);
    SelectObject(memDC, memBM);
    SelectObject(memDC, GetStockObject(DC_BRUSH));
    SetDCBrushColor(memDC, RGB(0, 0, 0)); // афарбування певної області
    Rectangle(memDC, 0, 0, 700, 700);
}

```

```

    SelectObject(memDC, GetStockObject(DC_BRUSH));
    SetDCBrushColor(memDC, RGB(0, 0, 0)); // 0-афарбування певної області
    SelectObject(memDC, GetStockObject(DC_PEN));
    Ellipse(memDC, Round_X - 15, Round_Y - 15, Round_X + 15, Round_Y + 15); //
Побудова самого кола
    BitBlt(dc, 0, 0, rct.right - rct.left, rct.bottom - rct.top, memDC, 0, 0,
SRCCOPY);
    DeleteDC(memDC);
    DeleteObject(memBM);
}
//-----
LRESULT WndProc(HWND hwnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    if(message == WM_DESTROY) // 0-авершення програми в разі закриття вікна
    {
        PostQuitMessage(0);
    }
    else if(message == WM_LBUTTONDOWN) // Пристосування для клацання мишкою
    {
        int Position_X = LOWORD(lparam);
        int Position_Y = HIWORD(lparam);
        double lengthX = abs(Round_X - Position_X);
        double lengthY = abs(Round_Y - Position_Y);
        double Distance = sqrt(lengthX*lengthX + lengthY*lengthY); //
Розрахунок відстані до шуканої області для визначення тональності
        Beep(50000 / Distance, 500); // Подання звукового сигналу
        if (Distance <= 13) // Якщо користувач знаходить місце розташування
шуканої області вікно закривається
        {
            ShowWindow(hwnd, SW_HIDE);
            End_Game();
        }
    }
    else if(message == WM_SIZE)
    {
        GetClientRect(hwnd, &rct);
    }
    else
    {
        return DefWindowProcA(hwnd, message, wParam, lParam);
    }
}
//-----
void Game() {
    WNDCLASSA wcl;
    memset(&wcl, 0, sizeof(WNDCLASSA));
    wcl.lpszClassName = "Sound Game"; // Назва вікна
    wcl.lpfnWndProc = WndProc; // Функція для роботи з вікном
    RegisterClassA(&wcl);
    HWND hwnd;
    hwnd = CreateWindow("Sound Game", "Sound Game", WS_OVERLAPPEDWINDOW, 20, 20,
700, 700, NULL, NULL, NULL, NULL);
    HDC dc = GetDC(hwnd);
    ShowWindow(hwnd, SW_SHOWNORMAL);
    MSG msg;
    Generation();
    while (1) {
        if (PeekMessageA(&msg, NULL, 0, 0, PM_REMOVE))
        {
            if (msg.message == WM_QUIT) break; // 0-авершення при натисканні
Есейпу
            DispatchMessage(&msg);
        }
        else

```

```
        {  
            WinShow(dc);  
        }  
    }  
}  
//-----  
int main()  
{  
    Zastavka();  
    return 0;  
}  
//-----
```



# Тестування

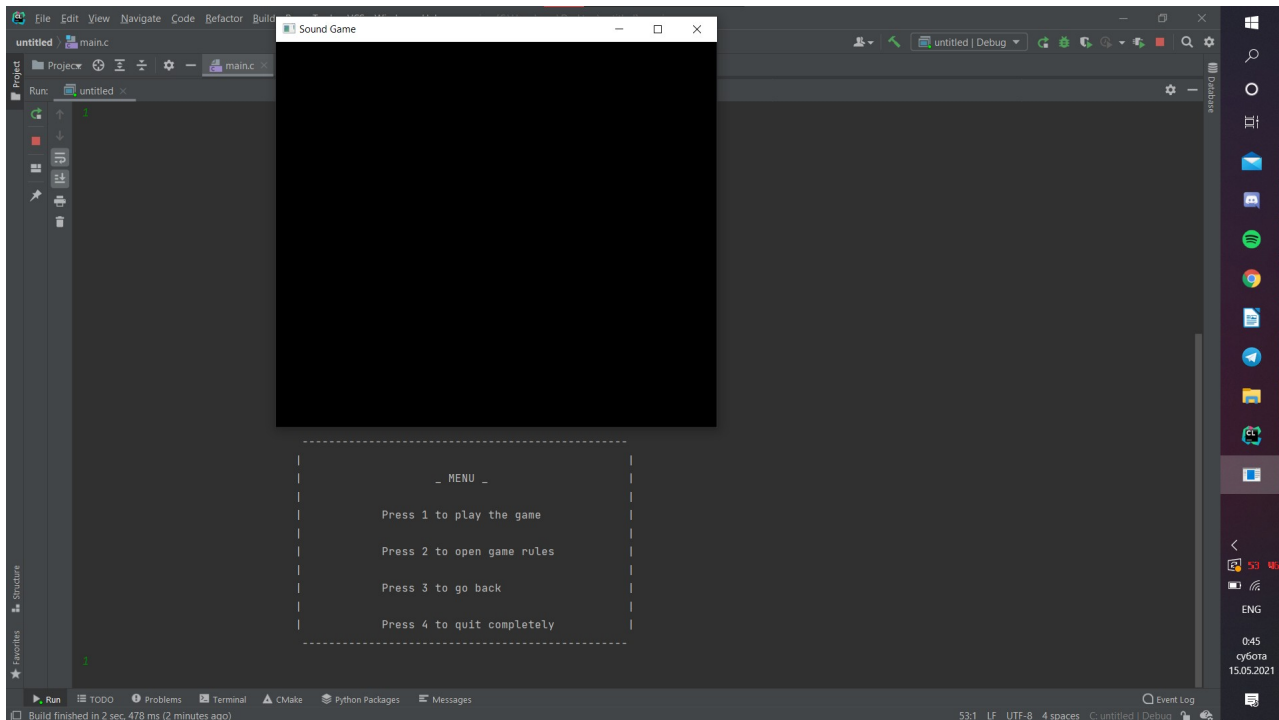
## Вивід заставки



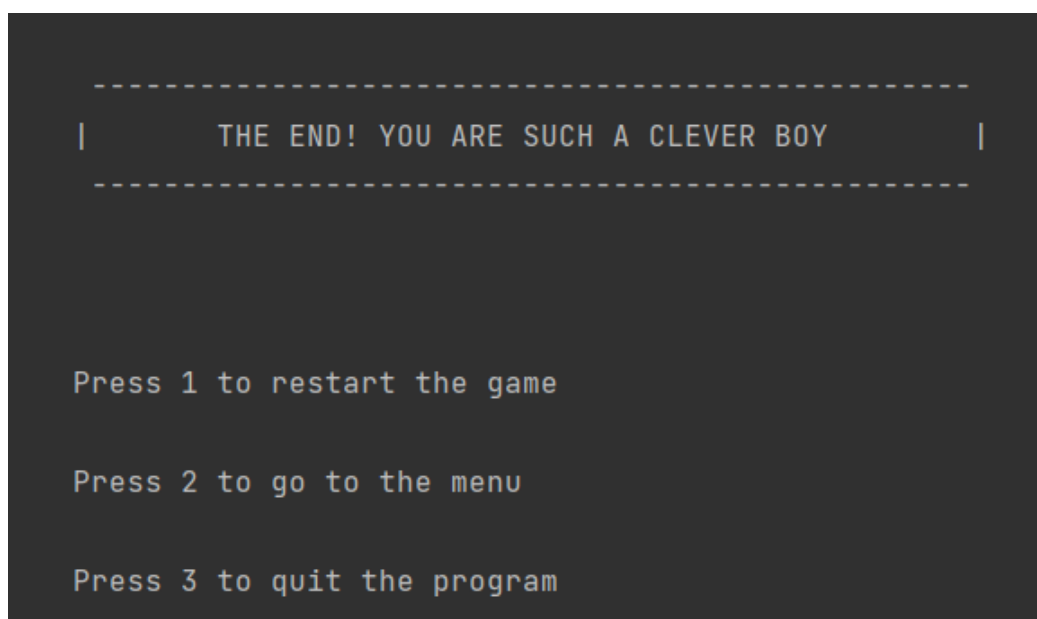
## Вивід меню



Поява нового вікна з грою, при цьому консоль залишається для виводу повідомлення про перемогу



Вивід повідомлення про перемогу з варіантами розвитку наступних подій



Вивід повідомлення про помилку, при неправильному натисканні клавіші

```
-----  
|          CHOOSE THE CORRECT BUTTON !!!          |  
-----
```

Вивід правил гри та варіантів розвитку наступних подій

```
-----  
|          Hi! In this game you need to          |  
|          look for hidden circle in the          |  
|          window with help of the sound.         |  
|          When the game start, use your          |  
|          mouse to click on Sound Game           |  
|          window. Be calm, because if you        |  
|          click too much, program will           |  
|          reply with a sum of sounds you         |  
|          clicked. Don't be scare of it!         |  
|          You should click every 1,5 seconds     |  
|          If you reach the hide circle you       |  
|          will win the game. For more            |  
|          information you can contact            |  
|          write a creator to his mail            |  
|                      Good luck!                  |  
-----  
  
Press 1 to go to the menu  
  
Press 2 to quit the program
```