

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

**Факультет прикладної математики
Кафедра системного програмування і спеціалізованих
комп’ютерних систем**

РОЗРАХУНКОВА-ГРАФІЧНА РОБОТА

з дисципліни

“Програмування-2.Програмування мовою С”

ТЕМА: “ Створення ігрової програми ”

Група: КВ-14

Виконав: Овчінніков Д.С.

Оцінка:

Київ – 2022

Постановка задачі:

Створити ігрову програму мовою програмування C.

Розробка і реалізація ігрових програм має вестися з врахуванням графічних та звукових можливостей, що надаються конкретним комп'ютером.

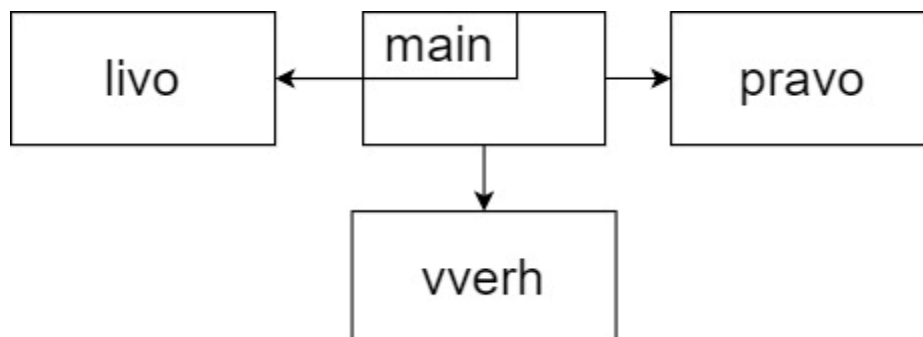
Програма мусить коректно розв'язувати поставлену задачу. Логічно відокремлені частини алгоритма реалізувати за допомогою окремих функцій.

Також потрібно передбачити та забезпечити виконання всіх можливих розгалужень алгоритма, 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.

Завдання за варіантом №14

«Міст». Даний міст з арками різної ширини. В нижній частині екрана розташований м'яч, яким можна керувати: рухати м'яч по рядку, зупиняти в потрібній позиції рядка і катити до мосту. Бали нараховуються, якщо м'яч проходить через арку.

Структура виклику функцій



livo - метод що викликається при русі вліво,здвигає позицію м'яча в матриці на 1 вліво та плавно переміщає його на екрані відповідно до матриці

pravo - метод що викликається при русі вправо,здвигає позицію м'яча в матриці на 1 вправо та плавно переміщає його на екрані відповідно до матриці

vverh - метод що викликається при русі вгору, плавно переміщує м'яч до верхнього кінця поля та підраховує кількість зароблених гравцем очок, при попаданні в арку м'яч через неї проходить та нараховує очки(велика арка -1,середня-2,маленька-3),у випадку потряпання в стіну врізається в неї і не нараховує очок.

Код програми

main.cpp

```
#include <SFML/Graphics.hpp>
#include <time.h>
#include <unistd.h>
#include <conio.h>
using namespace sf;

RenderWindow window(VideoMode(768, 768), "The Game!");
int blockWidht = 64;
Texture texture;
Texture ball;
Texture stage;
Sprite sprite[13];
Sprite sprite_ball;
Sprite sprite_stage0;
int grid[12][12] = { 0 };
int dx=6, dy=11, tmp;
int dxtmp=0, dytmp=0;
int stag=0;
int speed = 8;
int score = 0;
int proc = 1;
void livo()//метод що викликається при русі вліво
{
    if (dx != 0)//аби не вийти за межі масиву
    {
        tmp = grid[dy][dx];//зміщення м'яча в матриці
        grid[dy][dx]=grid[dy][dx-1];
        grid[dy][dx-1]=tmp;
        for (int i = 0; i <= blockWidht; i += speed)
        {
            window.clear(Color::White);
            sprite_ball.setPosition(dx*blockWidht-i, dy*blockWidht
); //переміщення м'яча
            window.draw(sprite_ball); //відмалювання
            for (int j = 0; j < 13; j++)
            {
                sprite[j].setPosition(j*blockWidht, 0 ); //встановлення спрайтів
                window.draw(sprite[j]); //відмалювання
            }
        }
    }
}
```

```

        window.display();
        usleep(75000);
    }
    dx--; //переміщуємо вліво на 1 пункт матриці вліво(64 пікселі)
}
}
void pravo() //метод що викликається при русі вправо
{
    if (dx != 11) //аби не вийти за межі масиву
    {
        tmp = grid[dy][dx]; //зміщення м'яча в матриці
        grid[dy][dx]=grid[dy][dx+1];
        grid[dy][dx+1]=tmp;
        for (int i = 0; i <= blockWidht; i += speed)
        {
            window.clear(Color::White);
            sprite_ball.setPosition(dx*blockWidht+i, dy*blockWidht
); //переміщення м'яча
            window.draw(sprite_ball); //відмалювання
            for (int j = 0; j < 13; j++)
            {
                sprite[j].setPosition(j*blockWidht, 0 ); //встановлення спрайтів
                window.draw(sprite[j]); //відмалювання
            }
            window.display();
            usleep(75000);
        }
        dx++; //переміщуємо вправо на 1 пункт матриці вправо(64 пікселі)
    }
}
void vverh() //метод що викликається при русі вгору
{
    while(proc==1) //цикл що виконується поки м'яч не дійде до кінця
    {
        if (dy == 1) //коли м'яч дійде до передостанньої клітинки він перевіряє"
скільки очок він отримає
        {
            if(grid[dy-1][dx]==4) //якщо 4 то це стінка отже гравець нічого не
отримує
            {
                //printf("You missed");
                speed=0; //швидкість 0 аби воно врізалось в стіну
                stag++; //перехід на наступну стадію
                proc=0; //закінчення та вихід з циклу
                break;
            }
            if(grid[dy-1][dx]==3) //якщо 3 то це найменша арка отже гравець от-
римує 3 очка
            {
                score+=3; //збільшуємо рахунок на 3
                //printf("You scored 3 points");
                dxtmp=5, dytmp=11; //використовуємо ці змінні аби м'яч не пригнуп
при прорисовці
            }
            if(grid[dy-1][dx]==2) //якщо 2 то це середня арка отже гравець отри-
мує 2 очка
            {
                score+=2; //збільшуємо рахунок на 2
                //printf("You scored 2 points");
                dxtmp=5, dytmp=11; //використовуємо ці змінні аби м'яч не пригнуп
при прорисовці
            }
        }
    }
}

```

```

    }
    if(grid[dy-1][dx]==1)//якщо 1 то це найбільша арка отже гравець от-
римує 1 очка
    {
        score+=1;//збільшуємо рахунок на 1
        //printf("You scored 1 points");
        dxtmp=5,dytmp=11;//використовуємо ці змінні аби м'яч не пригнуп
при прорисовці
    }
}
for (int i = 0; i <= blockWidht; i += speed)
{
    window.clear(Color::White);
    sprite_ball.setPosition(dx*blockWidht,dy*blockWidht-i
); //переміщення м'яча
    window.draw(sprite_ball);//відмалювання
    for (int j = 0; j < 13; j++)
    {
        sprite[j].setPosition(j*blockWidht,0 );//встановлення спрайтів
стіни та арок
        window.draw(sprite[j]);//відмалювання
    }
    window.display();
    usleep(30000);
}
dy--;//переміщуємо вгору на 1 пункт матриці вгору(64 пікселі)
if (dxtmp>0)//якщо м'яч попав в арку
{
    dxtmp=0;
    dytmp=0;
    proc=0;//вихід з циклу
    stag++;//перехід на наступну стадію
}
}
}

int main()
{
    ball.loadFromFile("img/ball.png");//присвоюємо м'ячу його текстуру
    stage.loadFromFile("img/main.png");//присвоюємо екрану взаємодії головний
екран
    // Логическое представление игрового поля
    srand(time(NULL));
    sprite_stage0.setTexture(stage);//присвоюємо екрану спрайт взаємодії на голов-
ному екрані
    sprite_ball.setTexture(ball);//присвоюємо м'ячу його спрайт

    while (window.isOpen())
    {
        Event event;
        while (window.pollEvent(event))
        {
            if (event.type == Event::Closed)
                window.close();
        }
        window.clear(Color::White);
        if (stag==0)//при запуску
        {
            sprite_stage0.setPosition(0,0);//виставляємо позицію спрайту на початок
            window.draw(sprite_stage0);
            window.display();//відмалювання

```

```

window.clear(Color::White);
proc=1;//повертаємо значення прос аби в грі працювало переміщення вгору
dx=rand()%12;//присвоюємо випадкове значення dx
speed=8;//присвоюємо випадкове значення швидкості
dy=11;//присвоюємо значення ду
usleep(75000);
}
    else if (stag==1)//після запуску виводить номер теперішнього кидка
    {
        texture.loadFromFile("img/var1.png");//загружаєм текстуру для першого
рівня
        sprite_stage0.setTexture(stage);
        stage.loadFromFile("img/sprobal.png");//присвоєння текстури виводу пер-
шої спроби
        sprite_stage0.setPosition(0,0);
        window.draw(sprite_stage0);
        window.display();
        window.clear(Color::White);
        proc=1;//анулювання прос аби можна було рухатись вгору
        dx=rand()%12;//присвоюємо випадкове значення dx
        speed=8;//присвоюємо випадкове значення швидкості
        dy=11;//присвоюємо значення ду
        for (int j = 0; j < 13; j++)//виставляння частин арки та стін на екрані
        {
            sprite[j].setTexture(texture);
            sprite[j].setTextureRect(IntRect(j*blockWidht,0 , blockWidht,
blockWidht));
        }
        for (int i = 0;i < 2;i++)//присвоюємо матриці значення стін,арок,м'яча
        {
            grid[0][i] = 4;
        }
        for (int i = 2;i < 5;i++)
        {
            grid[0][i] = 1;
        }
        for (int i = 5;i < 7;i++)
        {
            grid[0][i] = 4;
        }
        for (int i = 7;i < 9;i++)
        {
            grid[0][i] = 2;
        }
        grid[0][9] = 4;
        grid[0][10] = 3;
        grid[0][11] = 4;
        grid[11][dx] = 8;
    }
    else if (stag==2)//вивід екрану гри
    {
        for (int j = 0; j < 13; j++)
        {
            sprite[j].setPosition(j*blockWidht,0 );
            window.draw(sprite[j]);
        }
        sprite_ball.setPosition(dx*blockWidht,dy*blockWidht );
        window.draw(sprite_ball);
        window.display();
        window.clear(Color::White);
    }
    else if (stag==3)

```

```

{
    texture.loadFromFile("img/var2.png");//загрузка текстури 2 рівня
    Sprite sprite[13];
    sprite_stage0.setTexture(stage);
    stage.loadFromFile("img/sproba2.png");//загрузка текстур виводу 2 спроби
    sprite_stage0.setPosition(0,0);
    window.draw(sprite_stage0);
    window.display();
    window.clear(Color::White);
    proc=1;
    dx=rand()%12;
    speed=8;
    dy=11;
    for (int j = 0; j < 13; j++)//загрузка текстур арок та стін для 2 рівня
    {
        sprite[j].setTexture(texture);
        sprite[j].setTextureRect(IntRect(j*blockWidht,0 , blockWidht,
blockWidht));
    }
    grid[0][0] = 4;//присвоєння матриці значення згідно з рівнем та розта-
шуванням арок та стін
    for (int i = 1;i <= 3;i++)
    {
        grid[0][i] = 1;
    }
    for (int i = 4;i <= 5;i++)
    {
        grid[0][i] = 4;
    }
    grid[0][6] = 3;
    for (int i = 7;i <= 9;i++)
    {
        grid[0][i] = 4;
    }

    for (int i = 10;i <= 11;i++)
    {
        grid[0][i] = 2;
    }
    grid[11][dx]=8;
}
else if (stag==4)
{
    for (int j = 0; j < 13; j++)
    {
        sprite[j].setPosition(j*blockWidht,0 );
        window.draw(sprite[j]);
    }
    sprite_ball.setPosition(dx*blockWidht,dy*blockWidht );
    window.draw(sprite_ball);
    window.display();
    window.clear(Color::White);
}
else if (stag==5)
{
    texture.loadFromFile("img/var3.png");
    Sprite sprite[13];
    sprite_stage0.setTexture(stage);
    stage.loadFromFile("img/sproba3.png");
    sprite_stage0.setPosition(0,0);
    window.draw(sprite_stage0);
    window.display();
}

```

```

window.clear(Color::White);
proc=1;
dx=rand()%12;
speed=8;
dy=11;
for (int j = 0; j < 13; j++)
{
    sprite[j].setTexture(texture);
    sprite[j].setTextureRect(IntRect(j*blockWidht,0 , blockWidht,
blockWidht));
}
grid[0][0] = 4;
grid[0][1] = 3;
grid[0][2] = 4;
for (int i = 3; i <= 4; i++)
{
    grid[0][i] = 2;
}
for (int i = 5; i <= 7; i++)
{
    grid[0][i] = 4;
}
for (int i = 8; i <= 10; i++)
{
    grid[0][i] = 1;
}
grid[0][11] = 4;
grid[11][dx]=8;
}
else if (stag==6)
{
    for (int j = 0; j < 13; j++)
    {
        sprite[j].setPosition(j*blockWidht,0 );
        window.draw(sprite[j]);
    }
    sprite_ball.setPosition(dx*blockWidht,dy*blockWidht );
    window.draw(sprite_ball);
    window.display();
    window.clear(Color::White);
}
else if (stag==7)//кінець,вибір того яку текстуру вивести в залежності від
кількості очок
{
    if (score == 0)
    {
        stage.loadFromFile("img/score0.png");
    }
    else if (score == 1)
    {
        stage.loadFromFile("img/score1.png");
    }
    else if (score == 2)
    {
        stage.loadFromFile("img/score2.png");
    }
    else if (score == 3)
    {
        stage.loadFromFile("img/score3.png");
    }
    else if (score == 4)
    {

```



```

        stage.loadFromFile("img/score4.png");
    }
    else if (score == 5)
    {
        stage.loadFromFile("img/score5.png");
    }
    else if (score == 6)
    {
        stage.loadFromFile("img/score6.png");
    }
    else if (score == 7)
    {
        stage.loadFromFile("img/score7.png");
    }
    else if (score == 8)
    {
        stage.loadFromFile("img/score8.png");
    }
    else if (score == 9)
    {
        stage.loadFromFile("img/score9.png");
    }

    sprite_stage0.setPosition(0,0);
    window.draw(sprite_stage0);
    window.display();
    window.clear(Color::White);
    usleep(75000);
}
if (event.type == sf::Event::KeyPressed)//обробка дій якщо нажали кнопку
{
    if ((stag==2) || (stag==4) || (stag==6))//під час гри
    {
        if (event.key.code == sf::Keyboard::Left)//якщо нажали ліву стрілку
        то виконуємо функцію вліво
        {
            livo();
        }

        if (event.key.code == sf::Keyboard::Right)//якщо нажали вправо
        стрілку то виконуємо функцію вправо
        {
            pravo();
        }

        if (event.key.code == sf::Keyboard::Up)//якщо нажали вгору стрілку
        то виконуємо функцію вгору
        {
            vverh();
        }
    }
    if ((stag==0) || (stag==1) || (stag==3) || (stag==5) || (stag==7))//під час
    екранів взаємодії
    {
        if (event.key.code == sf::Keyboard::LShift)
        {
            stag++;//на наступний етап
            usleep(75000);
        }
    }
}


```

```

    }
    //printf("TOTAL SCORE %d\n",score);
    return 0;
}

```

Результат

 The Game!

— □ ×

РГР з програмування

14 варіант

«Міст». Даний міст з арками різної ширини. В нижній частині екрана розташований м'яч, яким можна керувати: рухати м'яч по рядку, зупиняти в потрібній позиції рядка і котити до мосту.

Бали нараховуються, якщо м'яч проходить через арку.

Чим менший розмір арки тим більше очок за її проходження ви отримаєте



Використовуйте цю клавішу для переміщення вліво



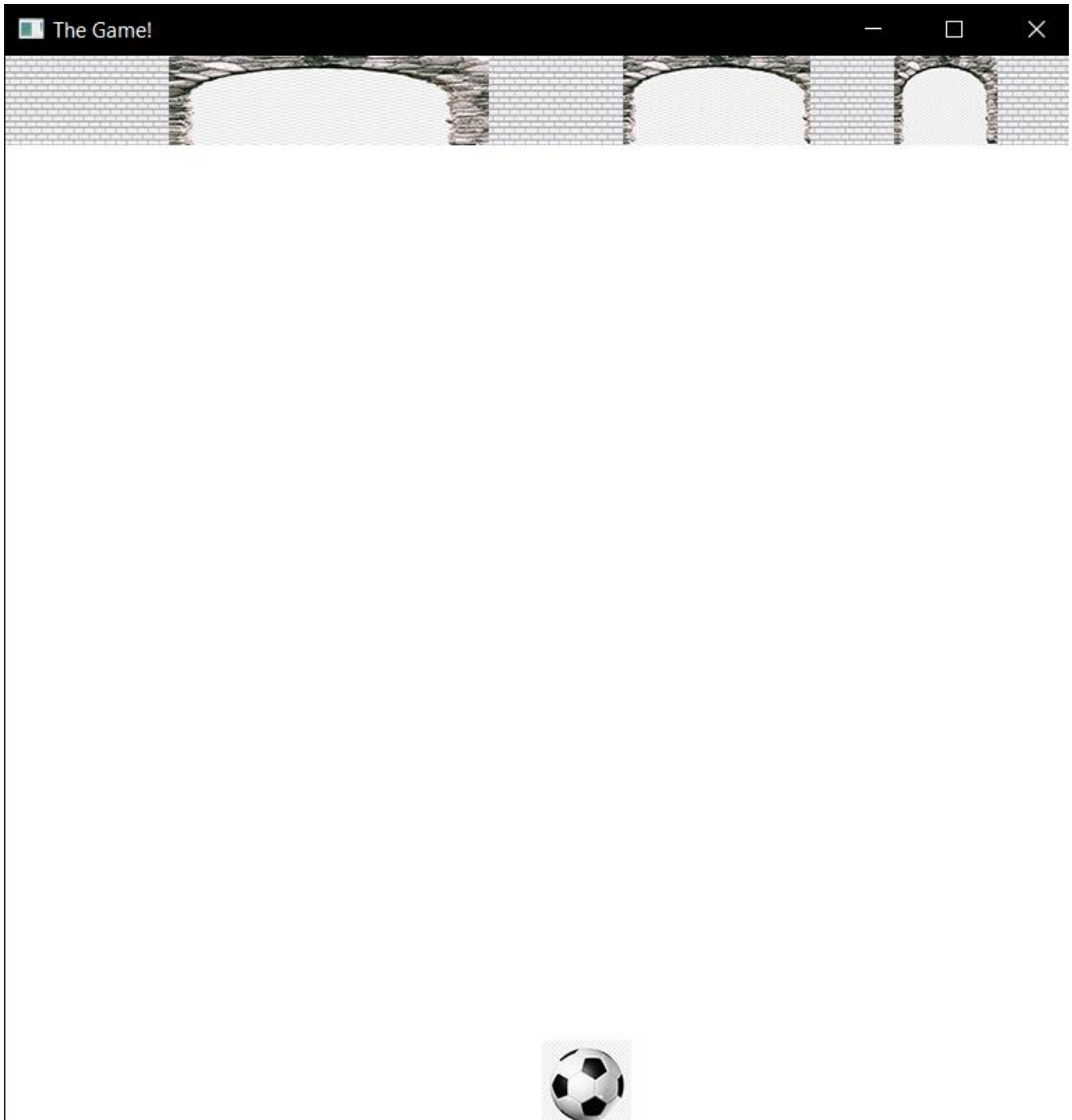
Використовуйте цю клавішу для переміщення вправо

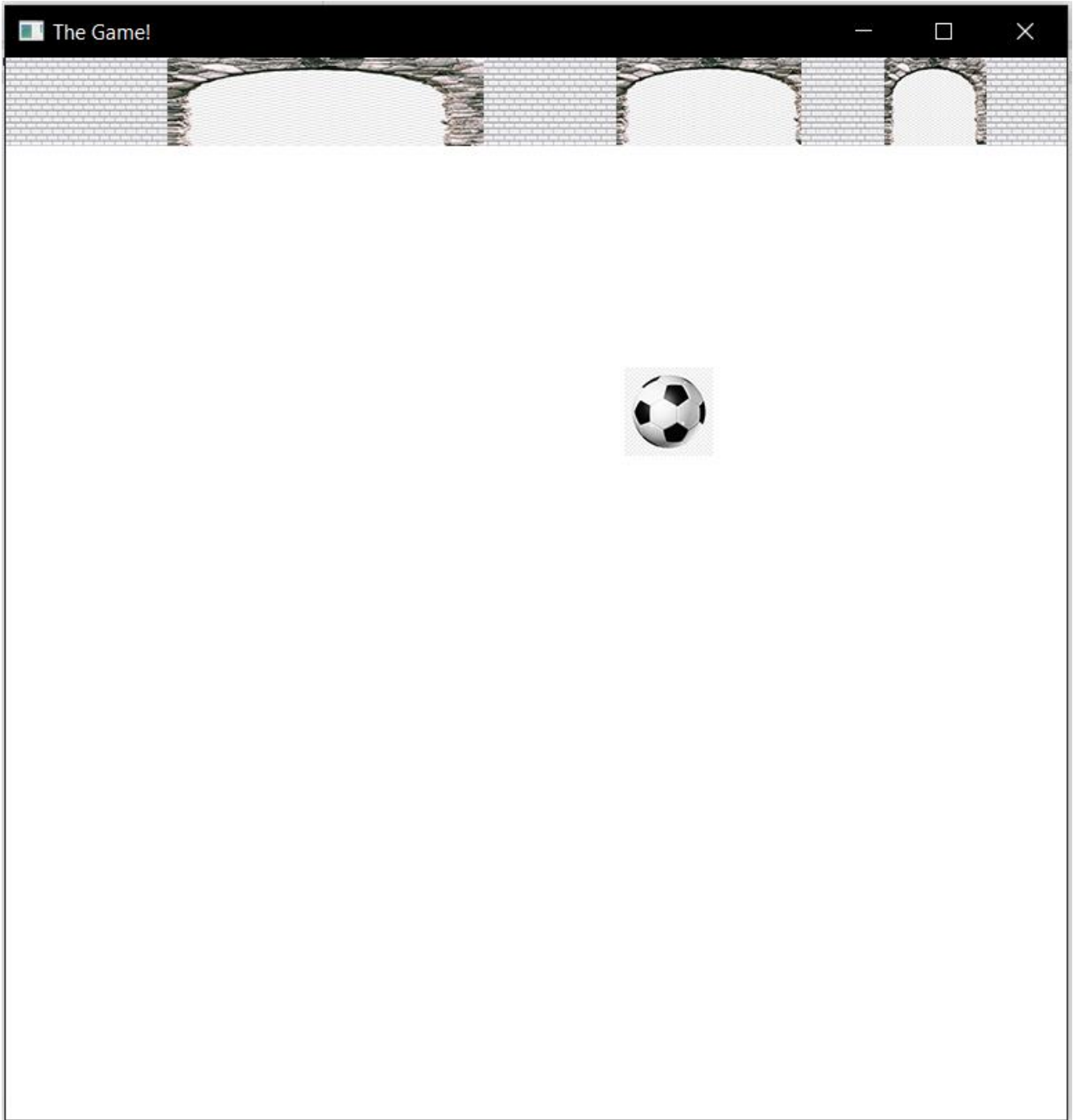


Коли виберете потрібний рядок натисніть цю клавішу і покотіть м'яч, зупинити його буде не можна

У вас буде 3 спроби,аби продовжити натисніть LShift

1 КИДОК
LShift аби продовжити





Ваш результат : 3