

**Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки**

Лабораторна робота №4
з дисципліни
«Алгоритми і структури даних»

Виконав:

Студент групи ІМ-42

Лобань Михайло Юрійович

номер у списку групи: 21

Перевірила:

Молчанова А. А.

Київ 2024

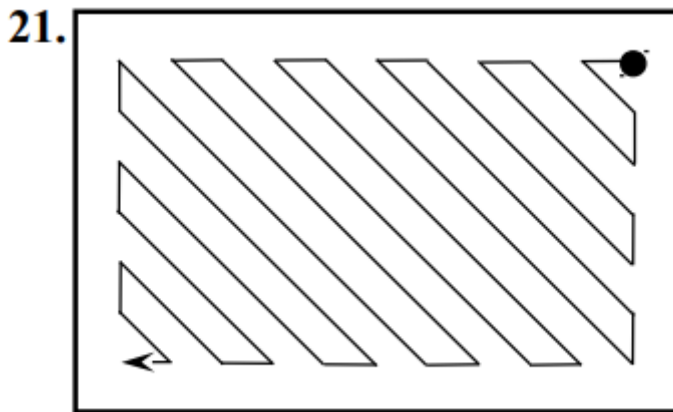
Постановка задачі

1. Оскільки екран монітора має площинні координати так само, як і двовимірний масив (матриця), але, на відміну від останнього, надає можливість візуально спостерігати виконання способу обходу, дана лабораторна робота виконується в координатах екрану монітора (дивись методичні вказівки до виконання даної лабораторної роботи).

2. Завданням даної лабораторної роботи є виконання заданого за варіантом способу обходу на екрані монітору в текстовому режимі, проставляючи довільний символ клавіатури (наприклад ‘*’) у порядку заданого способу обходу.

3. Оскільки при виводі символу у правий ніжній кут екрану відбувається зсув зображення на один рядок вгору (якщо тільки не використовується прямий доступ до відеопам’яті), *останній рядок екрану монітора при виконанні завдання заповнювати не треба.*

Завдання для варіанту



Текст програми

```
#include <windows.h>
```

```
#include <stdio.h>
```

```
int main() {
```

```
    HANDLE hout = GetStdHandle(STD_OUTPUT_HANDLE);
```

```
CONSOLE_SCREEN_BUFFER_INFO csbi;
```

```
GetConsoleScreenBufferInfo(hout, &csbi);
```

```
int width = csbi.srWindow.Right - csbi.srWindow.Left;
```

```
int height = csbi.srWindow.Bottom - csbi.srWindow.Top;
```

```
char character = '1';
```

```
int delay = 1;
```

```
int x_up = width;
```

```
int y_up = 0;
```

```
int x_down = width - 1;
```

```
int y_down = 0;
```

```
int direction = -1;
```

```
int symbol_counter = 0;
```

```
int x0, y0;
```

```
int diagonals_counter = 0;
```

```
COORD pos;
```

```
while (diagonals_counter <= width + height) {
```

```
    if (direction < 0) {
```

```
        x0 = x_up;
```

```
        y0 = y_up;
```

```
        while (x0 >= 0 && x0 <= width && y0 >= 0 && y0 <= height) {
```

```
            pos.X = x0;
```

```
            pos.Y = y0;
```

```
            SetConsoleCursorPosition(hout, pos);
```

```
            printf("%c", character);
```

```
            Sleep(delay);
```

```

    x0 += direction;
    y0 += direction;
}
if (height % 2 != 0) {
    if (y_up < height && y_up != height - 1) {
        y_up+=2;
    } else if(y_up == height - 1) {
        y_up++;
        x_up--;
    } else {
        x_up-=2;
    }
} else {
    if (y_up < height) {
        y_up+=2;
    } else {
        x_up-=2;
    }
}
} else {
x0 = x_down;
y0 = y_down;
while (x0 >= 0 && x0 <= width && y0 >= 0 && y0 <= height) {
    pos.X = x0;
    pos.Y = y0;
    SetConsoleCursorPosition(hout, pos);
    printf("%c", character);
    Sleep(delay);
}
}

```

```

    x0 += direction;
    y0 += direction;
}

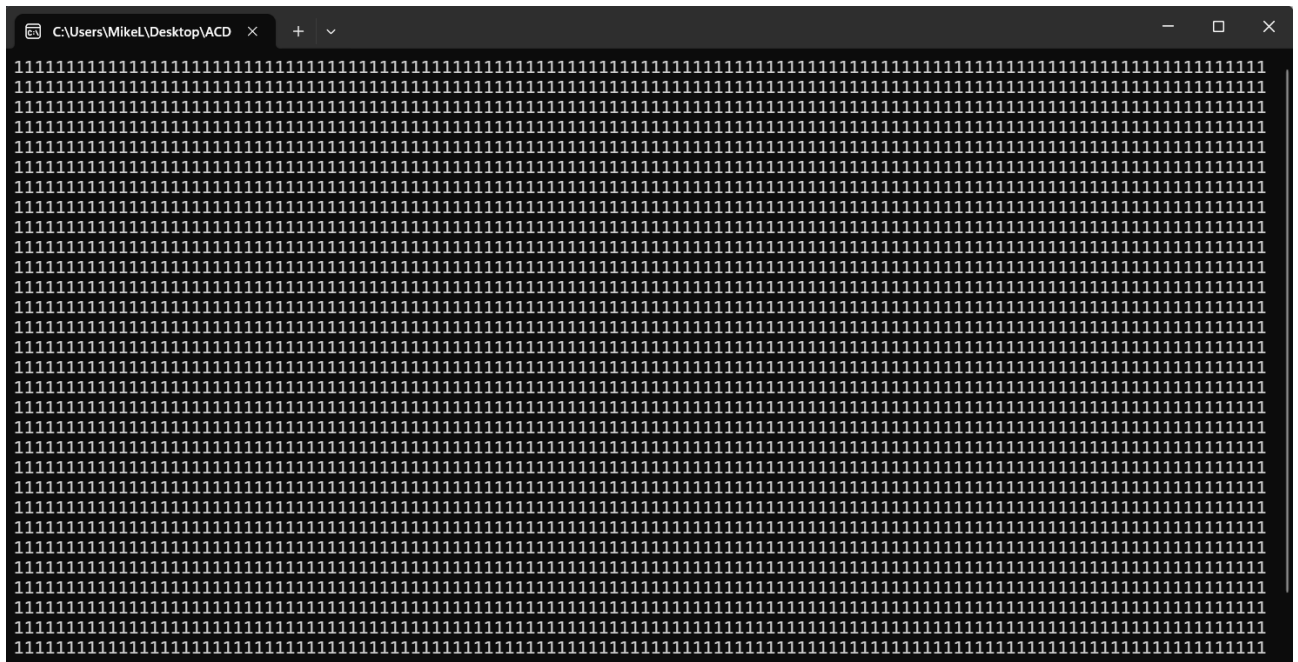
if (width % 2 != 0) {
    if (x_down > 0) {
        x_down-=2;
    } else {
        y_down+=2;
    }
} else {
    if (x_down > 0 && x_down != 1) {
        x_down-=2;
    } else if (x_down == 1) {
        x_down--;
        y_down++;
    } else {
        y_down+=2;
    }
}

direction = -direction;
diagonals_counter++;
}

return 0;
}

```

Результати тестування



Висновки

В ході виконання даної лабораторної роботи було продемонстровано та реалізовано один із варіантів способу обходу координат екрану монітора у текстовому режимі з використанням заданого символу. Завдяки такому підходу, було забезпечено візуальне відображення проходження по координатах екрана, що дозволяє простежити правильність виконання обраного алгоритму обходу.