УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчет по лабораторной работе №7.2

по предмету «Основы алгоритмизации и программирования»

Вариант 18

Выполнил:

Машевский Д.В.

Гр. 351003

Проверил:

Данилова Г. В.

Минск 2024

**Задание:**

Разработать программу, которая генерирует магический квадрат  
чётно-нечётной степени (6, 10, 18, …).

**Код программы Delphi:**

Unit FormsLaba73Unit7;

Interface

Uses

Winapi.Windows,

Winapi.Messages,

System.SysUtils,

System.Variants,

System.Classes,

Vcl.Graphics,

Vcl.Controls,

Vcl.Forms,

Vcl.Dialogs,

Vcl.Menus,

Vcl.Imaging.Jpeg,

Vcl.ExtCtrls,

Vcl.Grids,

Vcl.StdCtrls,

Vcl.Buttons;

Const

MAX\_DIGIT = 26;

Type

TArr = Array Of Array Of Integer;

Type

TMainForm = Class(TForm)

DataEdit: TEdit;

EnterBitBtn: TBitBtn;

MainMenu: TMainMenu;

PopupMenu: TPopupMenu;

RoflImage: TImage;

NFile: TMenuItem;

NOpenFile: TMenuItem;

NSaveFile: TMenuItem;

NDeveloper: TMenuItem;

DataStringGrid: TStringGrid;

GayImage: TImage;

SquareLabel: TLabel;

Label1: TLabel;

Procedure EnterBitBtnClick(Sender: TObject);

Procedure CheckEnter(Sender: TObject; Var Key: Char);

Procedure DataEditKeyPress(Sender: TObject; Var Key: Char);

Procedure NDeveloperClick(Sender: TObject);

Procedure NOpenFileClick(Sender: TObject);

Private

{ Private declarations }

Public

{ Public declarations }

End;

Var

MainForm: TMainForm;

Implementation

{$R \*.dfm}

Procedure BuildMagicSquareEven(N: Integer; Var Matrix: TArr);

Var

I, J, Num: Integer;

Begin

//Заполнение квадрата числами

Num := 1;

I := 0;

J := N Div 2;

Repeat

Matrix[I][J] := Num;

Inc(Num);

Dec(I);

Inc(J);

If (I < 0) Then

I := N - 1;

If (J >= N) Then

J := 0;

If (Matrix[I][J] <> 0) Then

Begin

Inc(I, 2);

Dec(J);

If (I >= N) Then

I := I - N;

If (J < 0) Then

J := N - 1;

End;

Until (Num > N \* N);

End;

Procedure BuildMagicSquareOdd(N: Integer; Var Matrix: TArr);

Var

I, J, Num: Integer;

Begin

//Инициализация матрицы нулями

For I := 0 To N - 1 Do

For J := 0 To N - 1 Do

Matrix[I][J] := 0;

//Начальные координаты

I := N Div 2;

J := N - 1;

//Заполнение квадрата числами

For Num := 1 To N \* N Do

Begin

Matrix[I][J] := Num;

//Новые координаты

I := (I - 1 + N) Mod N;

J := (J + 1) Mod N;

//Проверка на занятость клетки

If Matrix[I][J] <> 0 Then

Begin

I := (I + 1) Mod N;

J := (J - 2 + N) Mod N;

End;

End;

End;

Function CheckZeroStr(Str: String): Boolean;

Var

I, Len: Integer;

Fl: Boolean;

Begin

Fl := False;

Len := Length(Str) - 1;

For I := 1 To Len Do

Begin

If (Str[I] = '0') And (I <> High(Str)) Then

Begin

If (Str[I + 1] <> '') Then

Fl := True;

End;

End;

CheckZeroStr := Fl;

End;

Procedure TMainForm.CheckEnter(Sender: TObject; Var Key: Char);

Var

Chars: Set Of Char;

EditText: String;

Begin

EditText := (Sender As TEdit).Text;

If (Length(EditText) = 0) Then

Begin

Chars := ['0' .. '9', #8, #31];

End

Else

Chars := ['0' .. '9', #8, #31];

If Not(Key In Chars) Then

Key := #0;

End;

Procedure TMainForm.DataEditKeyPress(Sender: TObject; Var Key: Char);

Var

Value: Integer;

Begin

CheckEnter(Sender, Key);

If Key = #8 Then

If (CheckZeroStr(DataEdit.Text)) Then

DataEdit.Text := '';

If TryStrToInt(DataEdit.Text + Key, Value) Then

If (Value > MAX\_DIGIT) Then

Key := #0;

If DataEdit.Text = '0' Then

Begin

DataEdit.Text := '';

End;

End;

Procedure TMainForm.EnterBitBtnClick(Sender: TObject);

Var

N, I, J: Integer;

MagicSquare: TArr;

Begin

If DataEdit.Text <> '' Then

Begin

N := StrToInt(DataEdit.Text);

DataStringGrid.RowCount := N;

DataStringGrid.ColCount := N;

SetLength(MagicSquare, N, N);

If (N Mod 2 = 0) Then

BuildMagicSquareEven(N, MagicSquare)

Else

BuildMagicSquareOdd(N, MagicSquare);

//Вывод квадрата на экран

For I := 0 To N - 1 Do

For J := 0 To N - 1 Do

DataStringGrid.Cells[J, I] := IntToStr(MagicSquare[I][J]);

End;

End;

Procedure TMainForm.NDeveloperClick(Sender: TObject);

Var

MessageBoxCaption: String;

MessageBoxText: String;

IsMessageShow: Boolean;

Begin

IsMessageShow := False;

If Not IsMessageShow Then

Begin

IsMessageShow := True;

MessageBoxCaption := 'О разработчике';

MessageBoxText := 'Машевский Даниил Витальевич, группа 351003, Лабораторная №

7.3';

MessageBox(Handle, PChar(MessageBoxText), PChar(MessageBoxCaption), MB\_OK);

End;

IsMessageShow := False;

End;

Function FindLeng(Path: String): Integer;

Var

Len, Value: Integer;

InputFile: TextFile;

Begin

AssignFile(InputFile, Path);

Reset(InputFile);

Len := 0;

While Not(Eof(InputFile)) Do

Begin

Read(InputFile, Value);

Inc(Len);

End;

FindLeng := Len;

End;

Procedure TMainForm.NOpenFileClick(Sender: TObject);

Var

OpenDialog: TOpenDialog;

InputFile: TextFile;

NumArr: TArr;

Values: String;

Len, I, J: Integer;

Begin

OpenDialog := TOpenDialog.Create(Nil);

OpenDialog.Title := 'Выберите файл';

OpenDialog.Filter := 'Текстовые файлы (\*.txt)|\*.txt';

If OpenDialog.Execute Then

Begin

AssignFile(InputFile, OpenDialog.FileName);

Reset(InputFile);

Readln(InputFile, Len);

DataEdit.Text := IntToStr(Len);

CloseFile(InputFile);

End;

End;

End.

**Код программы на Java:**

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class MagicSquare {

// Метод для построения четно-нечетного магического квадрата

private static int[][] buildOddEvenMagicSquare(int n) {

int[][] magicSquare = new int[n][n];

int subSquareSize = n / 2 + 1;

// Заполнение каждой части магическим квадратом нечетного порядка

int[][] subSquare = buildOddMagicSquare(subSquareSize);

for (int i = 0; i < n / 2; i++) {

for (int j = 0; j < n / 2; j++) {

magicSquare[i][j] = subSquare[i][j];

magicSquare[i + n / 2][j + n / 2] = subSquare[i][j];

magicSquare[i][j + n / 2] = subSquare[i][j] + (n / 2) \* (n / 2);

magicSquare[i + n / 2][j] = subSquare[i][j] + (n / 2) \* (n / 2) \* 3;

}

}

// Обмен элементов в верхнем правом и нижнем левом квадрате

for (int i = 0; i < n / 2; i++) {

for (int j = 0; j < n / 2; j++) {

if (i < n / 2 && j >= n / 2) {

int temp = magicSquare[i][j];

magicSquare[i][j] = magicSquare[i + n / 2][j - n / 2];

magicSquare[i + n / 2][j - n / 2] = temp;

}

}

}

// Обмен элементов в верхнем левом и нижнем правом квадрате

for (int i = 0; i < n / 2; i++) {

for (int j = 0; j < n / 2; j++) {

if (i >= n / 2 && j < n / 2) {

int temp = magicSquare[i][j];

magicSquare[i][j] = magicSquare[i - n / 2][j + n / 2];

magicSquare[i - n / 2][j + n / 2] = temp;

}

}

}

return magicSquare;

}

// Метод для построения магического квадрата нечетного порядка

private static int[][] buildOddMagicSquare(int n) {

int[][] magicSquare = new int[n][n];

int num = 1;

int i = 0;

int j = n / 2;

while (num <= n \* n) {

magicSquare[i][j] = num;

num++;

int newi = (i - 1 < 0) ? n - 1 : i - 1;

int newj = (j + 1 >= n) ? 0 : j + 1;

if (magicSquare[newi][newj] != 0) {

i = (i + 1 >= n) ? 0 : i + 1;

} else {

i = newi;

j = newj;

}

}

return magicSquare;

}

public static void main(String[] args) throws FileNotFoundException {

Scanner scanner = new Scanner(System.in);

int conf;

int order = 0;

String path;

System.out.print("Введите порядок магического квадрата: ");

System.out.println("1 - файл, 0 - консоль");

conf = scanner.nextInt();

if (conf == 1) {

System.out.println("Enter the file path");

path = scanner.next();

try {

File file = new File(path);

Scanner fileScanner = new Scanner(file);

order = fileScanner.nextInt();

fileScanner.close();

} catch (FileNotFoundException e) {

System.out.println("File not found");

}

} else {

order = scanner.nextInt();

}

{

int[][] magicSquare = buildOddEvenMagicSquare(order);

System.out.println("Магический квадрат порядка " + order + ":");

for (int i = 0; i < order; i++) {

for (int j = 0; j < order; j++) {

System.out.print(magicSquare[i][j] + "\t");

}

System.out.println();

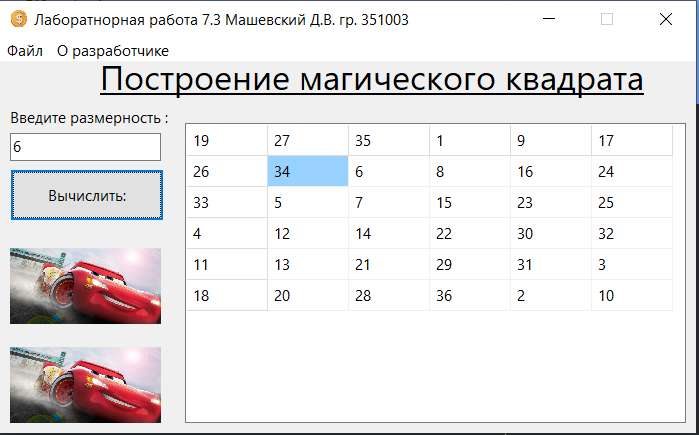
}

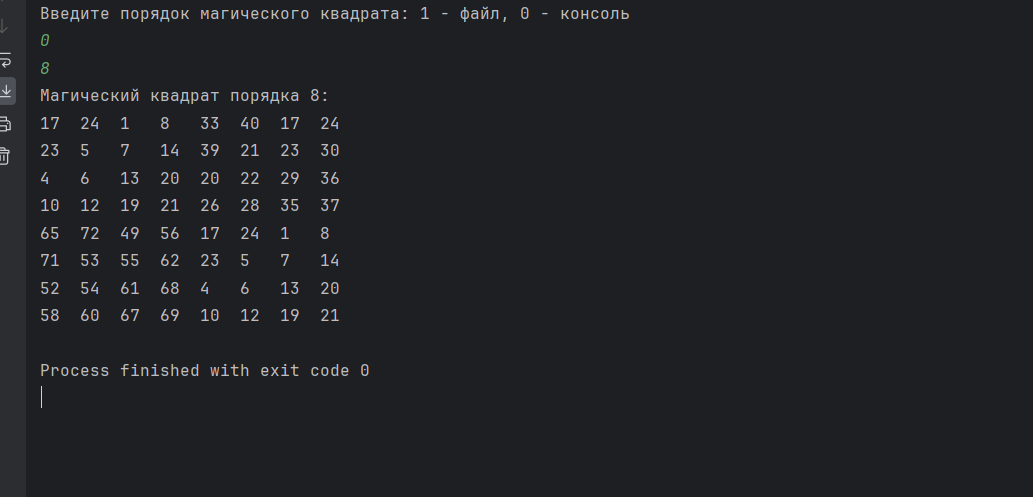
}

}

}

**Скриншоты:**





**Блок-схема:**

