УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчет по лабораторной работе №2.1

по предмету

Основы алгоритмизации и программирования

Вариант 14

Выполнил:

Крутько А.А.

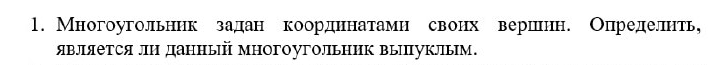
Проверила:

Данилова Г.В.

Группа 251004

Минск 2022

Задание:



Код программы на **Delphi**:

Program LabFirstBlockSecond;

Uses

System.SysUtils, Math;

Var

N, I, J: Integer;

CrossLine1, CrossLine2: Real;

Abscisses, Ordinates: Array of Real;

IsIncorrect, IsTrue: Boolean;

Const

MIN\_NUM = 3;

Begin

Repeat

Write('Введите число сторон у многоугольника — N (N > 2): ');

IsIncorrect := True;

Try

Readln(N);

Except

IsIncorrect := False;

Writeln('Проверьте правильность ввода');

End;

If (IsIncorrect) and (N < MIN\_NUM) Then

Begin

IsIncorrect := False;

Writeln('Проверьте правильность ввода');

End;

Until IsIncorrect;

SetLength(Abscisses, N + 3);

SetLength(Ordinates, N + 3);

For I := 1 to N do

Begin

Repeat

IsIncorrect := True;

Write('Abscissa[',I,']= ');

Try

Readln(Abscisses[I]);

Except

IsIncorrect := False;

Writeln('Проверьте правильность ввода');

End;

Until IsIncorrect;

End;

For I := 1 to N do

Begin

Repeat

IsIncorrect := True;

Write('Ordinate[',I,']= ');

Try

Readln(Ordinates[I]);

Except

IsIncorrect := False;

Writeln('Проверьте правильность ввода');

End;

Until IsIncorrect;

End;

Abscisses[0] := Abscisses[N];

Ordinates[0] := Ordinates[N];

Abscisses[N + 1] := Abscisses[1];

Abscisses[N + 2] := Abscisses[2];

Ordinates[N + 1] := Ordinates[1];

Ordinates[N + 2] := Ordinates[2];

IsTrue := True;

I := 1;

While (I <= N) and (IsTrue) Do

Begin

CrossLine1 := (Abscisses[I - 1] - Abscisses[I]) \* (Ordinates[I + 1] -   
 Ordinates[I]) - (Ordinates[I - 1] - Ordinates[I]) \*   
 (Abscisses[I + 1] - Abscisses[I]);

CrossLine2 := (Abscisses[I] - Abscisses[I + 1]) \* (Ordinates[I + 2] -  
 Ordinates[I + 1]) - (Ordinates[I] - Ordinates[I + 1]) \*   
 (Abscisses[I + 2] - Abscisses[I + 1]);

If CrossLine1 \* CrossLine2 < 0 Then

IsTrue := False

Else

Inc(I);

End;

If IsTrue Then

Write('Выпуклый')

Else

Write('Не выпуклый');

Readln;

End.

Код программы на **C++**:

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

int n, k;

float crossLine1, crossLine2;

float \*abscisses, \*ordinates;

bool isIncorrect, isTrue;

const int MIN\_NUM = 3;

do

{

isIncorrect = false;

cout << "Введите число сторон многоугольника N, N > 2: ";

cin >> n;

if (cin.fail())

{

cin.clear();

while (cin.get() != '\n');

isIncorrect = true;

cout << "Проверьте правильность ввода" << endl;

}

if (!isIncorrect && cin.get() != '\n')

{

cin.clear();

while (cin.get() != '\n');

cout << "Проверьте правильность ввода" << endl;

isIncorrect = true;

}

if (!isIncorrect && n < MIN\_NUM)

{

isIncorrect = true;

cout << "Проверьте правильность ввода" << endl;

}

} while (isIncorrect);

abscisses = new float[n+3];

ordinates = new float[n+3];

for (int i = 1; i <= n; ++i)

{

do {

isIncorrect = false;

cout << "Abscissa[" << (i) << "]: ";

cin >> abscisses[i];

if (cin.fail())

{

cin.clear();

while (cin.get() != '\n');

isIncorrect = true;

cout << "Проверьте правильность ввода" << endl;

}

if (!isIncorrect && cin.get() != '\n')

{

cin.clear();

while (cin.get() != '\n');

cout << "Проверьте правильность ввода" << endl;

isIncorrect = true;

}

} while (isIncorrect);

}

for (int i = 1; i <= n; ++i)

{

do {

isIncorrect = false;

cout << "Ordinate[" << (i) << "]: ";

cin >> ordinates[i];

if (cin.fail())

{

cin.clear();

while (cin.get() != '\n');

isIncorrect = true;

cout << "Проверьте правильность ввода" << endl;

}

if (!isIncorrect && cin.get() != '\n')

{

cin.clear();

while (cin.get() != '\n');

cout << "Проверьте правильность ввода" << endl;

isIncorrect = true;

}

} while (isIncorrect);

}

abscisses[0] = abscisses[n];

ordinates[0] = abscisses[n];

abscisses[n + 1] = abscisses[1];

abscisses[n + 2] = abscisses[2];

ordinates[n + 1] = ordinates[1];

ordinates[n + 2] = ordinates[2];

isTrue = true;

k = 1;

while (k <= n && isTrue)

{

crossLine1 = (abscisses[k - 1] - abscisses[k]) \* (ordinates[k + 1] -   
 ordinates[k]) - (ordinates[k - 1] - ordinates[k]) \*   
 (abscisses[k + 1] - abscisses[k]);

crossLine2 = (abscisses[k] - abscisses[k + 1]) \* (ordinates[k + 2] -   
 ordinates[k + 1]) - (ordinates[k] - ordinates[k + 1]) \*   
 (abscisses[k + 2] - abscisses[k + 1]);

if (crossLine1 \* crossLine2 < 0)

isTrue = false;

else

k++;

}

delete[] abscisses, ordinates;

if (isTrue)

cout << "Выпуклый";

else

cout << "Не выпуклый";

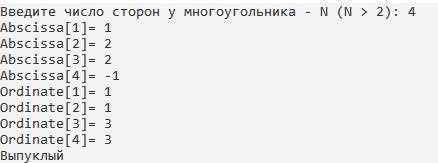
return 0;

}

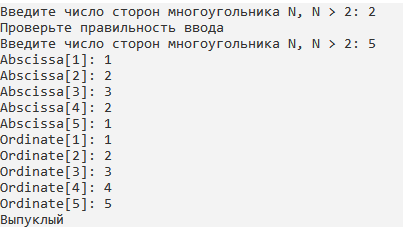
Код программы на **Java**:

import java.util.Scanner;  
public class LabFirstBlockSecond {  
 public static void main(String[] args) {  
 Scanner scan = new Scanner(System.in);  
 boolean isIncorrect, isTrue;  
 int n = 0, k = 1;  
 float[] abscisses, ordinates;  
 float crossLine1, crossLine2;  
 final int MIN\_NUM = 3;  
 do {  
 System.out.print("Введите число сторон многоугольника N, N > 2: ");  
 isIncorrect = false;  
 try {  
 n = Integer.parseInt(scan.nextLine());  
 } catch (Exception e) {  
 System.err.println("Проверьте правильность ввода данных");  
 isIncorrect = true;  
 }  
 if (!isIncorrect && n < MIN\_NUM) {  
 System.err.println("Проверьте правильность ввода данных в нужном   
 диапазоне");  
 isIncorrect = true;  
 }  
 } while(isIncorrect);  
 abscisses = new float[n + 3];  
 ordinates = new float[n + 3];  
 for (int i = 1; i <= n; ++i) {  
 do {  
 isIncorrect = false;  
 System.out.print("Abscissa[" + i + "]: ");  
 try {  
 abscisses[i] = Float.parseFloat(scan.nextLine());  
 } catch (Exception e) {  
 System.err.println("Проверьте правильность ввода данных");  
 isIncorrect = true;  
 }  
 } while(isIncorrect);  
 }  
 for (int i = 1; i <= n; ++i) {  
 do {  
 isIncorrect = false;  
 System.out.print("Ordinate[" + i + "]: ");  
 try {  
 ordinates[i] = Float.parseFloat(scan.nextLine());  
 } catch (Exception e) {  
 System.err.println("Проверьте правильность ввода данных");  
 isIncorrect = true;  
 }  
 } while (isIncorrect);  
 }  
 scan.close();  
 isTrue = true;  
 abscisses[0] = abscisses[n];  
 ordinates[0] = abscisses[n];  
 abscisses[n + 1] = abscisses[1];  
 abscisses[n + 2] = abscisses[2];  
 ordinates[n + 1] = ordinates[1];  
 ordinates[n + 2] = ordinates[2];  
 while (k <= n && isTrue)  
 {  
 crossLine1 = (abscisses[k - 1] - abscisses[k]) \* (ordinates[k + 1] -   
 ordinates[k]) - (ordinates[k - 1] - ordinates[k]) \*   
 (abscisses[k + 1] - abscisses[k]);  
 crossLine2 = (abscisses[k] - abscisses[k + 1]) \* (ordinates[k + 2] -   
 ordinates[k + 1]) - (ordinates[k] - ordinates[k + 1]) \*   
 (abscisses[k + 2] - abscisses[k + 1]);  
 if (crossLine1 \* crossLine2 < 0)  
 isTrue = false;  
 else  
 k++;  
 }  
 if (isTrue)  
 System.out.println("Выпуклый");  
 else  
 System.out.println("Не выпуклый");  
 }  
}

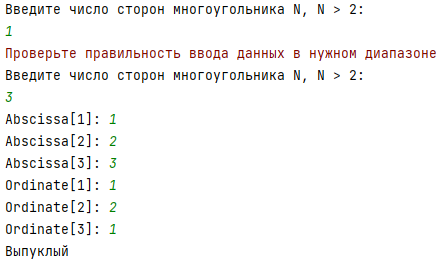
Результат на **Delphi**:



Результат на **C++**:

****

Результат на **Java**:



**Блок-cхема**:

