УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчет по лабораторной работе №1.2

по предмету «Основы алгоритмизации и программирования»

Вариант 31

Выполнил:

Чвей К.В.

Гр. 451003

Проверил:

Данилова Г. В.

Минск 2024

**Задание:**

Найти n-ый член ряда Фибоначчи, элементы которого вычисляются по формулам: a1=a2=1; ai=ai-1 + ai-2, (i>2).

**Код программы Delphi:**

Program Fibonacci;

Uses

System.SysUtils;

Var

ValueN, First, Result, Temp, I : Integer;

IsCorrect : Boolean;

Const

UPPER\_LIMIT\_INPUT : Integer = 45;

Begin

Writeln('Задание: найти n-ый член ряда Фибоначчи, элементы

которого вычисляются по формулам: a1=a2=1; ai=ai-1 +

ai-2, (i>2).');

First := 1;

Temp := 0;

Result := 0;

ValueN := 0;

IsCorrect := False;

While Not IsCorrect Do

Begin

WriteLn('Введите номер члена ряда Фибоначчи, не превышающее

45');

Try

Readln(ValueN);

IsCorrect := True;

Except

WriteLn('Неверный ввод, повторите попытку');

End;

If(ValueN > UPPER\_LIMIT\_INPUT) Then

Begin

WriteLn('Вы ввели слишком большое число, повторите

попытку');

IsCorrect := False;

End;

If(ValueN < 1) Then

Begin

WriteLn('Член ряда Фибоначчи с таким номером не

существует, повторите попытку');

IsCorrect := False;

End;

End;

For I := 1 To ValueN Do

Begin

Temp := Result + First;

First := Result;

Result := Temp;

End;

Writeln(Result);

Readln;

End.

**Код программы С++:**

#include <iostream>

int main()

{

setlocale(LC\_ALL, "ru-RU");

int valueN;

int first;

int result;

int temp;

int i;

const int UPPER\_LIMIT\_INPUT = 10000;

bool isCorrect;

first = 1;

temp = 0;

result = 0;

isCorrect = false;

std::cout << "Найти n-ый член ряда Фибоначчи, элементы

которого вычисляются по формулам: a1=a2=1; ai=ai-1

+ ai-2, (i>2).";

while (!isCorrect)

{

std::cout << "Введите номер члена ряда Фибоначчи, не

превышающее 10000";

std::cin >> valueN;

if (std::cin.fail() || valueN < 0 || valueN >

UPPER\_LIMIT\_INPUT)

{

std::cin.clear();

while (std::cin.peek() != '\n')

std::cin.ignore();

std::cout << "Введено не корректное число, повторите

ввод";

if (valueN < 0)

{

std::cout << "номер элемента не может быть

отрецательным, повторите ввод";

}

else if (valueN > UPPER\_LIMIT\_INPUT)

{

std::cout << "номер элемента не может превышать

10000, повторите ввод";

}

else

{

std::cout << "номер элемента должен быть числом,

повторите ввод";

}

}

else if (std::cin.peek() != '\n')

{

std::cout << "Введено некорректное число, повторите

попытку";

while (std::cin.peek() != '\n')

std::cin.ignore();

}

else

{

while (std::cin.peek() != '\n')

std::cin.ignore();

isCorrect = true;

}

}

for (i = 0; i < valueN; i++)

{

temp = result + first;

first = result;

result = temp;

}

std::cout << result;

}

**Код программы Java:**

import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 final int UPPER\_LIMIT\_INPUT = 10000;  
 int valueN;  
 int first;  
 int result;  
 int temp;

int i;  
 int value;  
 boolean isCorrect;  
 Scanner sc = new Scanner(System.in);  
 first = 1;  
 valueN = 0;  
 temp = 0;  
 result = 0;  
 isCorrect = false;  
 System.out.println("Найти n-ый член ряда Фибоначчи,

элементы которого вычисляются по

формулам: a1=a2=1; ai=ai-1 + ai-2,

(i>2).");  
 while (!isCorrect) {  
 System.out.println("Введите номер члена ряда Фибоначчи, не

превышающее 10000 ");  
 if (sc.hasNextInt()) {  
 value = sc.nextInt();  
 if (value < 0) {  
 isCorrect = false;  
 System.out.println("номер элемента не может быть

отрецательным, повторите

ввод");  
 sc.nextLine();  
 }  
 else if (value > UPPER\_LIMIT\_INPUT) {  
 isCorrect = false;  
 System.out.println("номер элемента не может

превышать 10000, повторите

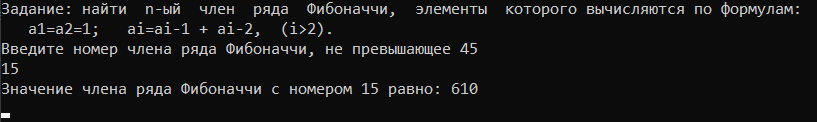
ввод");  
 sc.nextLine();  
 }  
 else {  
 valueN = value;  
 isCorrect = true;  
 }  
 }  
 else {  
 isCorrect = false;  
 System.out.println("номер элемента не может превышать

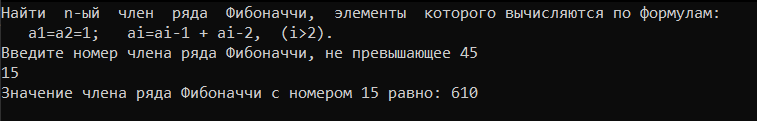
10000 или не являться числом,

повторите ввод ");  
 sc.next();  
 }  
 }  
 sc.close();  
 for (i = 0; i < valueN; i++) {  
 temp = result + first;  
 first = result;  
 result = temp;  
 }  
 System.out.println(result);  
 }  
}

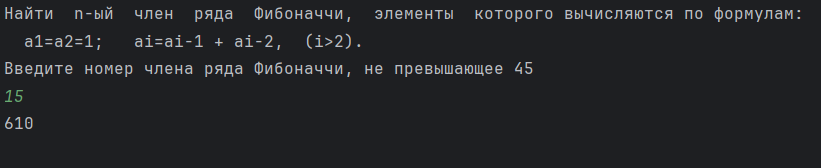
**Скриншоты:**

**Delphi:**



**C++:** 

**Java:**



**Блок-схема:**

