УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчет по лабораторной работе №2.2

по предмету «Основы алгоритмизации и программирования»

Вариант 20

Выполнил:

Захвей И.В.

Гр. 351005

Проверил:

Данилова Г. В.

Минск 2023

**Задание:**

Дано натуральное число n. Найти все меньшие n числа Мерсена. (Простое число называется числом Мерсена, если оно может быть представлено в виде 2р – 1, где р – тоже простое число.

**Код программы Delphi:**

Program Project2;

Uses

System.SysUtils;

Const

MAX\_VALUE = 3000000000;

MIN\_VALUE = 1;

Procedure PrintInf();

Begin

Writeln('Программа вычисляет все числа Марсена меньше n, где n [',

MIN\_VALUE, ', ', MAX\_VALUE, ']');

Writeln('Число Мерсена -- простое число, которое можно представить в виде 2^р – 1,

где р – тоже простое число.');

End;

Function InputNum(): FixedUInt;

Var

Number: FixedUInt;

IsCorrect: Boolean;

Begin

Repeat

IsCorrect := True;

Writeln('Введите n');

Try

Readln(Number);

Except

Writeln('Ошибка, не верный тип данных!');

IsCorrect := False;

End;

If (IsCorrect And (Number < MIN\_VALUE) Or (Number > MAX\_VALUE)) Then

Begin

IsCorrect := False;

Writeln('Ошибка, число должно быть больше ', MIN\_VALUE,

' и меньше ', MAX\_VALUE);

End;

Until IsCorrect;

InputNum := Number;

End;

Function IsNumSimpl(Numb: FixedUInt): Boolean;

Var

IsSimpl: Boolean;

NumbSqrt, I: Integer;

Begin

NumbSqrt := Trunc(Sqrt(Numb));

IsSimpl := True;

If (Numb > 3) Then

For I := 2 To NumbSqrt Do

If (Numb Mod I = 0) Then

IsSimpl := False;

IsNumSimpl := IsSimpl;

End;

Procedure PrintMersNum(HighBord: FixedUInt);

Var

IsBordIncross: Boolean;

MersenNum: FixedUInt;

I: Integer;

Begin

MersenNum := 1;

I := 2;

IsBordIncross := True;

While IsBordIncross Do

Begin

MersenNum := MersenNum \* 2 + 1;

IsBordIncross := MersenNum < HighBord;

If (IsBordIncross And IsNumSimpl(I) And IsNumSimpl(MersenNum)) Then

Writeln('Mersen(', I, ') -- ', MersenNum);

Inc(I);

End;

End;

Var

HighBorder: FixedUInt;

Begin

PrintInf;

HighBorder := InputNum();

PrintMersNum(HighBorder);

Readln;

End.

**Код программы С++:**

#include <iostream>

using std::cout;

using std::cin;

const unsigned int MAX\_VALUE = 3000000000;

const int MIN\_VALUE = 1;

void printInf()

{

cout << "Программа вычисляет все числа Марсена меньше n, где n [" << MIN\_VALUE <<

", " << MAX\_VALUE << "]\n";

cout << "Число Мерсена -- простое число, которое можно представить в виде 2^р – 1,

где р – тоже простое число.\n";

}

unsigned int inputNum()

{

unsigned int number;

bool isIncorrect;

do

{

isIncorrect = false;

cin >> number;

if (cin.fail())

{

cin.clear();

cout << "Ошибка, введите только число\n";

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

isIncorrect = true;

}

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

{

cin.clear();

cout << "Ошибка, введите только число\n";

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

isIncorrect = true;

}

if (!isIncorrect && (number > MAX\_VALUE || number < MIN\_VALUE))

{

cout << "Ошибка, число должно быть больше " << MIN\_VALUE << " и меньше "

<< MAX\_VALUE << '\n';

isIncorrect = true;

}

} while (isIncorrect);

return number;

}

bool isNumSimple(int numb)

{

float numbSqrt;

numbSqrt = sqrt(numb) + 1;

if (numb > 3)

for (int i = 2; i < numbSqrt; i++)

if (numb % i == 0)

return false;

return true;

}

void printMersNum(unsigned int highBorder)

{

bool isBordIncros;

int mersenNum, i;

mersenNum = 1;

i = 2;

isBordIncros = true;

while (isBordIncros)

{

mersenNum = mersenNum \* 2 + 1;

isBordIncros = mersenNum < highBorder;

if (isBordIncros && isNumSimple(i) && isNumSimple(mersenNum))

cout << "Mersen(" << i << ") -- " << mersenNum << '\n';

i++;

}

}

int main()

{

setlocale(0, "");

unsigned int hihgBorder;

printInf();

hihgBorder = inputNum();

printMersNum(hihgBorder);

return 0;

}

**Код программы Java:**

import java.util.Scanner;

public class Main {

static void printInf(int min,long max){

System.out.println("Программа вычисляет все числа Марсена меньше n, где n ["+ min +", "+ max +"]");

System.out.println("Число Мерсена -- простое число, которое можно представить в виде 2^р – 1, где р – тоже простое число.");

}

static long inputNum(int min, long max){

long number;

boolean isIncorrect;

number = 0L;

isIncorrect = false;

Scanner input = new Scanner(System.in);

System.out.println("Введите n");

do {

isIncorrect = false;

try {

number = Long.parseLong(input.nextLine());

} catch (NumberFormatException error) {

System.err.println("Ошибка, неверные данные");

isIncorrect = true;

}

if (!isIncorrect && number < min || number > max){

isIncorrect = true;

System.err.println("Ошибка, число должно быть больше "+ min +" и меньше "+ max);

}

} while (isIncorrect);

return number;

}

static boolean isNumSimpl(long numb) {

int sqrtNum;

sqrtNum = (int) Math.sqrt(numb) + 1;

if (numb > 3) {

for (int i = 2; i < sqrtNum; i++) {

if (numb % i == 0) {

return false;

}

}

}

return true;

}

static void printMersen(long highBord) {

long mersenNum;

boolean isBordIncros;

int i;

i = 2;

mersenNum = 1;

isBordIncros = mersenNum < highBord;

while (isBordIncros) {

mersenNum = mersenNum \* 2 + 1;

isBordIncros = mersenNum < highBord;

if (isBordIncros && isNumSimpl(i) && isNumSimpl(mersenNum)) {

System.out.println("Mersen(" + i + ") -- " + mersenNum);

}

i++;

}

}

public static void main(String[] args) {

final int MIN\_VALUE = 1;

final long MAX\_VALUE = 3000000000L;

long highBorder;

printInf(MIN\_VALUE, MAX\_VALUE);

highBorder = inputNum(MIN\_VALUE, MAX\_VALUE);

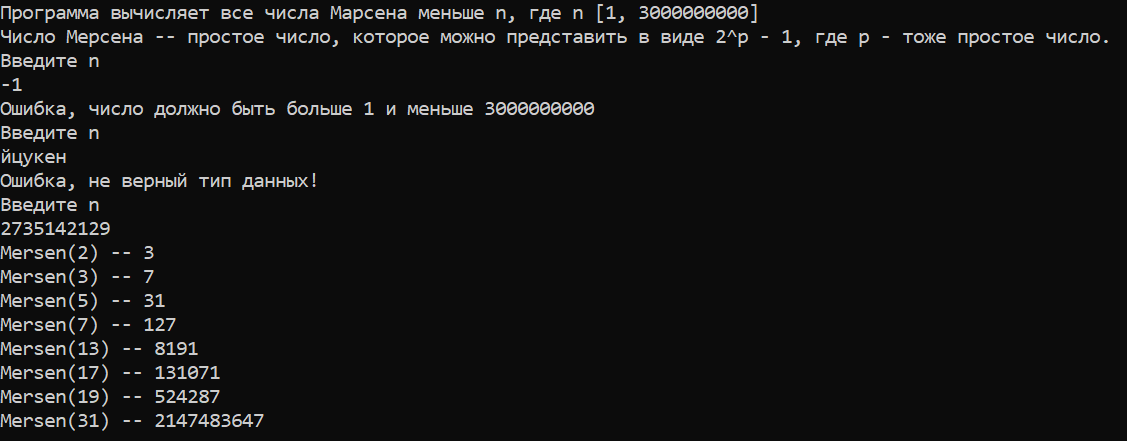
printMersen(highBorder);

}

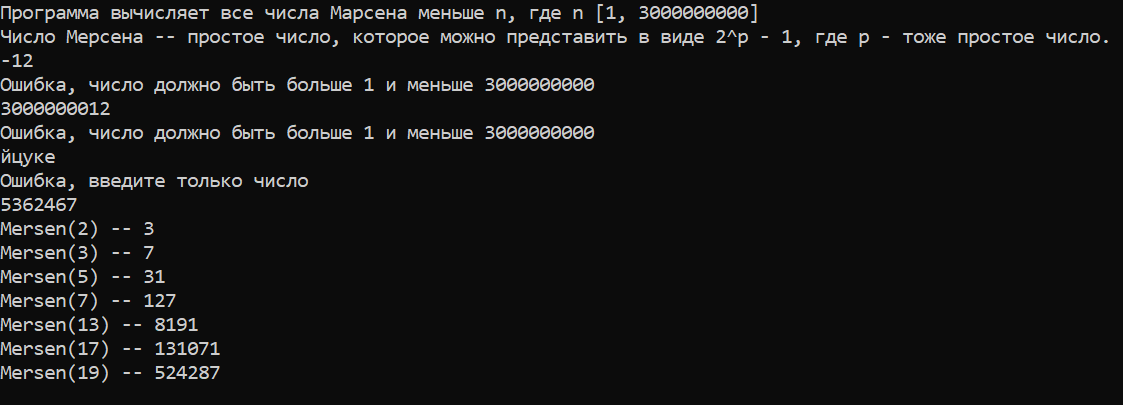
}

**Скриншоты:**

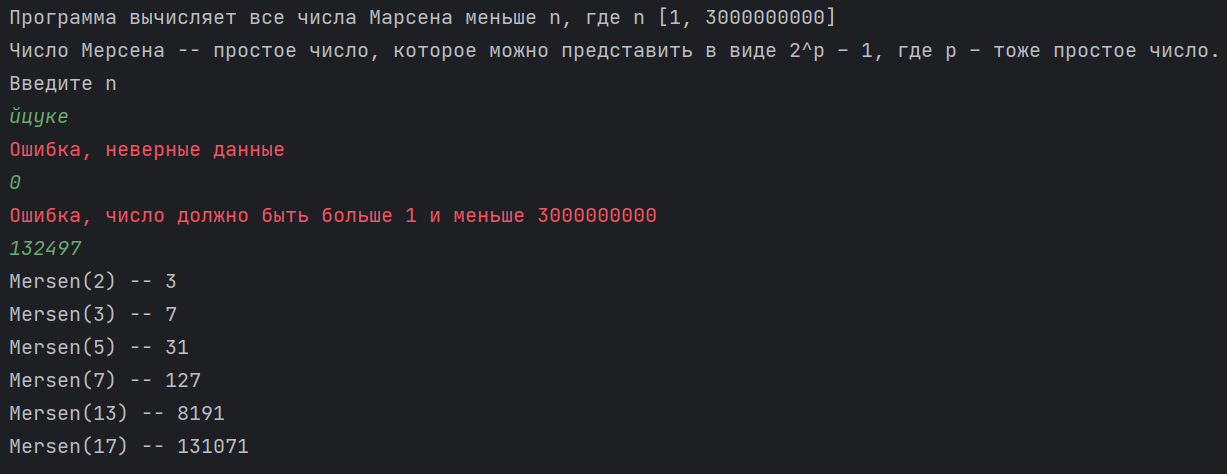
**Delphi:**

****

**C++:**

****

**Java:**

****

**Блок-схема:**



