

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«Омский государственный технический университет»
Кафедра «Информатика и вычислительная техника»

Отчёт по лабораторной работе № 1
по дисциплине
«Проектирование и тестирование программного обеспечения»

Выполнил:
Студент гр. ПИН-211
Сероухов Е.С. _____
(подп., дата)

Проверил:
Старший преподаватель каф. ИВТ
Карабцов Р.Д. _____
(подп., дата)

ВВЕДЕНИЕ

Цель работы: решить задачу, не требующую специальной подготовки.

Задача работы: Задача 1.1 (Задача $3n + 1$)

PC/UVa IDs: 110101/100

Рассмотрим следующий алгоритм генерации последовательности чисел. Начнем с целого числа n . Если n четно, то поделим на 2. Если n нечетно, то умножим на 3 и добавим 1. Будем повторять этот процесс с новым полученным n , пока n не станет равным 1. Например, для $n = 22$ будет сгенерирована следующая последовательность чисел:

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

Полагают (но это еще не доказано), что этот алгоритм сведется к $n = 1$ для любого целого n . По крайней мере, это предположение верно для всех целых чисел до 1 000 000.

Для данного n *длиной цикла n* будем называть число сгенерированных чисел до и включая 1. В примере, приведенном выше, длина цикла 22 равна 16. Для двух заданных чисел i и j вы должны определить максимальную длину цикла для всех чисел между i и j , включая обе конечные точки.

Входные данные

Входные данные будут состоять из серии пар целых чисел i и j , одна пара чисел в строке. Все целые числа будут меньше 1 000 000 и больше 0.

Выходные данные

Для каждой пары чисел i и j выведите i , j в том порядке, в каком они были введены, и после этого выведите максимальную длину цикла для всех целых чисел между i и j включая сами i и j . Эти три числа должны быть разделены одним пробелом, все три числа в одной строке, и для каждой строки входных данных должна быть одна строка выходных данных.

Пример входных данных

1 10

100 200

201 210

900 1000

Соответствующие выходные данные

1 10 20

100 200 125

201 210 89

900 1000 174

РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

1. Для решения задачи был написан код на ЯП Java в среде IntelliJ IDEA 2022.1.2.

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    final static Scanner in = new Scanner(System.in);
    public static final String ANSI_RED = "\u001B[31m";
    public static final String ANSI_RESET = "\u001B[0m";
    public static int[] three_N_plus_one(int n){
        List<Integer> list = new ArrayList<>(); // сюда пишем числа из
текущего цикла
        list.add(n); // добавляем первое число
        while (n!=1){ // пока не свелось к 1
повтор цикла
            if (n%2 == 0) // если четное то
                n/=2; // делим на 2
            else n=3*n+1; // иначе умножаем на 3 и
прибавляем 1
            list.add(n); // добавляем в лист
получившееся число
        }
        int[] array = list.stream().
            mapToInt(i->i).toArray(); // преобразование листа в
массив
        list.clear();
        return array; // возвращает массив (от n
до 1)
    }

    public static void IJ_generation_max(){
        int
            maxLength = 0,
            i = 0,
            j = 0;
        boolean fi = false,
            fj = false;

        // пока не будет введено значение в пределах (0; 1 000 000)
        // будет повторятся ввод
        while (!fi){
            System.out.print("Enter i: "); i = in.nextInt();
            if (i>0&&i<1000000)
                fi = true;
            else
                System.out.println(ANSI_RED+"retry (0 < i < 1 000
000)"+ANSI_RESET);
        }
        while (!fj){
            System.out.print("        j: "); j = in.nextInt();
            if (j>0&&j<1000000)
                fj = true;
            else
                System.out.println(ANSI_RED+"retry (0 < j < 1 000
000)"+ANSI_RESET);
        }
    }
}
```

```

        for (int n = i; n < j; n++) // перебираем все
числа от i до j
            if (three_N_plus_one(n).length > maxLength) // если длинна
цикла для текущего числа больше максимального
                maxLength = three_N_plus_one(n).length; // то длина
текущего цикла равна максимальной

        System.out.println("Max len: " + maxLength);
        System.out.println "[" + i + " " + j + " " + maxLength + " ]";
        System.out.println();
    }
    public static void main(String[] args) {
        for(;;) IJ_generation_max();
    }
}

```

2. Было проведено тестирование работы программы.

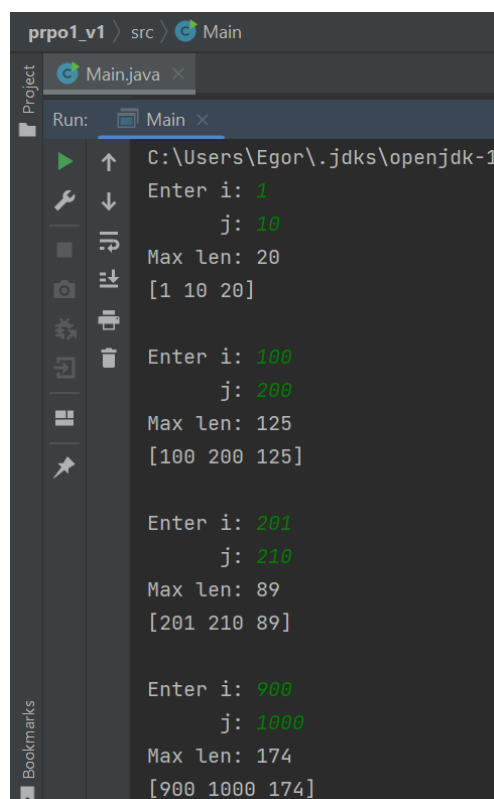


Рисунок 1 – Тестирование работы программы.

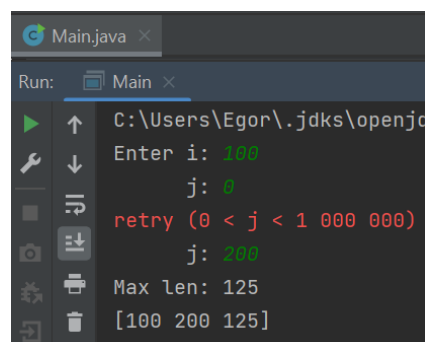


Рисунок 2 – Тестирование работы программы при некорректном вводе.

ЗАКЛЮЧЕНИЕ

Вывод: в ходе работы была написана программа на ЯП Java. Программа решила поставленную задачу и прошла проверку при стандартном тестировании, а также при тестировании с некорректными (по условию задачи) входными данными.