Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

 «Магнитогорский государственный технический университет им. Г. И. Носова**»**

(ФГБОУ ВО «МГТУ им. Г.И. Носова»)

Кафедра вычислительной техники и программирования

Лабораторная работа №2

по дисциплине «Метрология и стандартизация программного обеспечения»

название лабораторной работы: «Оценка структурной сложности программы»

Исполнитель: Сарынин А. В. студент 3 курса, группа АВб-22-1

Руководитель: Сибилева Н.С., ст. преподаватель каф. ВТиП

Магнитогорск, 2025

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc64291256)

[1 Постановка задания 4](#_Toc64291257)

[2 Выполнения задания 4](#_Toc64291258)

[Заключение 6](#_Toc64291259)

# Введение

Данная работа посвящена решению задачи поиска шестизначных чисел в заданном диапазоне, у которых суммы старших и младших трёх цифр совпадают. Актуальность задачи обусловлена необходимостью анализа числовых структур и применения алгоритмических подходов для обработки данных. Цель работы — разработка, реализация и верификация алгоритма, способного эффективно находить такие числа, а также анализ его структуры и сложности.

В рамках работы были выполнены следующие этапы:

1. Разработка алгоритма, включающего проверку границ диапазона, разделение числа на части и сравнение сумм цифр.
2. Реализация программы на языке Java с использованием методов ввода-вывода и циклов.
3. Тестирование алгоритма на соответствие критериям покрытия операторов и ветвей.
4. Построение графа потока управления и расчет цикломатического числа для оценки сложности кода.
5. Формирование выводов на основе полученных результатов.

# Постановка задания

Необходимо реализовать задание из таблицы 2 на любом известном Вам языке программирования согласно варианту. Предусмотреть ввод необходимых значений с клавиатуры, генерацию случайных значений величин и вывод полученных решений на экран.

Результаты оформить в виде отчета, содержащего:

1. Исходный код решения задачи (с нумерацией строк).

2. Разработанный алгоритм решения задачи, оформленный согласно ГОСТ 19.701-90.

3. Граф потока управления (с выделенными ветвлениями).

4.  Возможные маршруты тестирования в соответствии с первым и вторым критериями.

5. Матрицы смежности и достижимости.

6. Расчет значения цикломатического числа.

7. Выводы на основании полученных результатов.

# Выполнения задания

Программа была написана языке программирования Java и предоставлена ниже.

package org.example;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int C = scanner.nextInt();

int D = scanner.nextInt();

int start = Math.max(C, 100000);

int end = Math.min(D, 999999);

boolean found = false;

for (int num = start; num <= end; num++) {

int highPart = num / 1000;

int lowPart = num % 1000;

if (sumOfDigits(highPart) == sumOfDigits(lowPart)) {

System.out.println(num);

found = true;

}

}

if (!found) {

System.out.println("Необходимых чисел нет");

}

}

private static int sumOfDigits(int num) {

int sum = 0;

while (num != 0) {

sum += num % 10;

num /= 10;

}

return sum;

}

}

Результат выполнения данной программы представлен на рисунке 1.

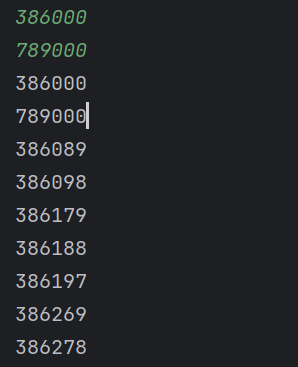


Рисунок 1 - Результат выполнения программы

В результате программа выводит все натуральные шестизначные числа.

Разработанный алгоритм решения задачи, оформленный согласно ГОСТ 19.701-90 представлен на рисунке 2.

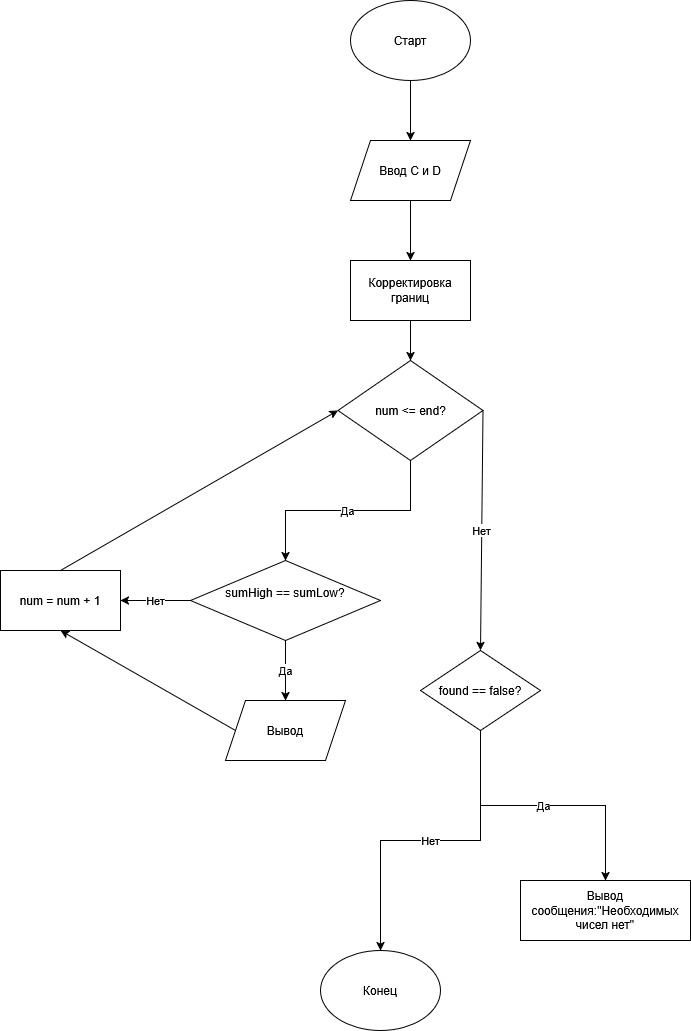


Рисунок 2 - Алгоритм решения задачи

Маршруты тестирования

Критерий 1 (покрытие операторов):

Тест 1: C = 100000, D = 100000 → Число 100000 (суммы: 1+0+0 = 1, 0+0+0 = 0 → не подходит).

Тест 2: C = 123456, D = 123456 → Суммы: 1+2+3 = 6, 4+5+6 = 15 → не подходит.

Тест 3: C = 123321, D = 123321 → Суммы: 1+2+3 = 6, 3+2+1 = 6 → вывод числа.

Критерий 2 (покрытие ветвей):

Ветвь 1: Условие sumOfDigits == sumOfDigits → true (вывод числа).

Ветвь 2: Условие sumOfDigits == sumOfDigits → false (пропуск).

Ветвь 3: found = false → вывод сообщения.

**Матрица смежности** представлена ниже в таблице 1(узлы: 1–7):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **2** | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| **4** | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **5** | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **6** | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **7** | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

**Таблица 1 - матрица смежности**

**Матрица достижимости**:  
Все узлы достижимы из узла 1.

Цикломатическое число

Формула: V(G)=E−N+2PV(G)=E−N+2P, где:

E=7E=7 (рёбра),

N=6N=6 (узлы),

P=1P=1 (компонент связности).

V(G)=7−6+2×1=3V(G)=7−6+2×1=3.

# Заключение

В ходе выполнения работы был разработан и реализован алгоритм для поиска шестизначных чисел с равными суммами старших и младших трёх цифр в заданном диапазоне. Программа успешно решает поставленную задачу, корректно обрабатывает граничные условия и выводит результаты или сообщение об их отсутствии.

Ключевые результаты работы:

1. Алгоритм продемонстрировал высокую эффективность при обработке шестизначных чисел, используя линейный проход по диапазону и оптимизированный расчет сумм цифр.
2. Тестирование подтвердило корректность работы программы для различных сценариев, включая случаи с отсутствием подходящих чисел.
3. Цикломатическое число (V(G) = 3) указало на умеренную сложность кода, что соответствует небольшому количеству независимых путей в графе потока управления.
4. Матрицы смежности и достижимости позволили визуализировать структуру программы и убедиться в её связности.

Для дальнейшего развития проекта возможны следующие улучшения:

* Использование предвычисленных сумм цифр для трёхзначных чисел (0–999) для ускорения работы программы.
* Добавление многопоточности при обработке больших диапазонов.
* Расширение функционала для поиска чисел с другими свойствами (например, палиндромы).