



**«Московский государственный технический университет
имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

О т ч е т

по лабораторной работе № 3

Название лабораторной работы: Программирование
циклического процесса. Типы циклов.

Дисциплина: Алгоритмизация и программирование

Студент гр. ИУ6-13Б

(Подпись, дата)

Д.М. Майоров

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

О.А. Веселовская

(И.О. Фамилия)

Москва, 2025

Цель работы: целью данной лабораторной работы является освоение программирования циклических процессов через практическое применение всех трех видов циклов (for, while, do-while) для решения алгоритмической задачи. Разработка навыков сравнительного анализа эффективности различных циклических конструкций и обоснования выбора наиболее рационального варианта для конкретной задачи.

Задание: разработать программу, которая находит все натуральные числа из диапазона [111, 999], состоящие только из нечетных цифр. Реализовать решение задачи с использованием всех трех видов циклов: for, while, do while.

Выполнение: проект программы (с использованием for) изображен на рисунке 1, текст программы изображен на рисунке 2.

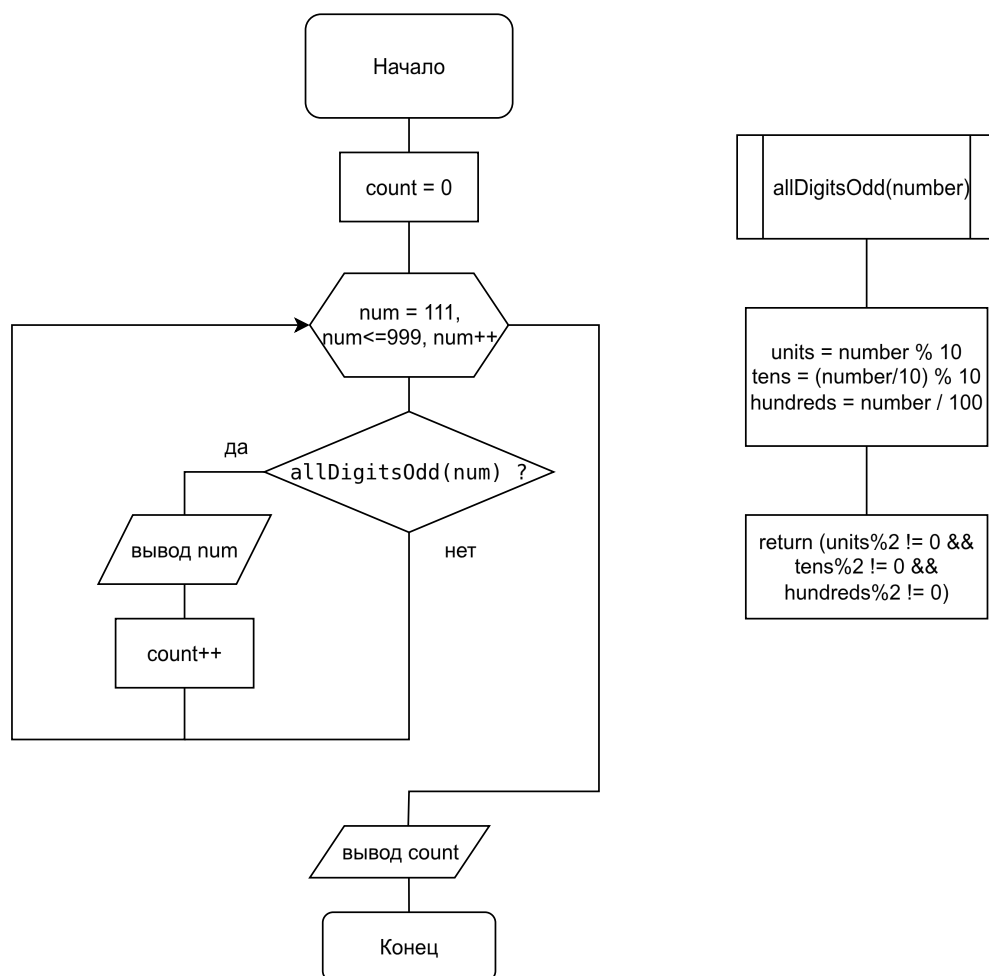


Рисунок 1 — Проект программы (с использованием for)

```

1  #include <iostream>
2  using namespace std;
3
4  bool allDigitsOdd(int number){
5      int units = number % 10;          // единицы
6      int tens = (number / 10) % 10;    // десятки
7      int hundreds = number / 100;     // сотни
8      return (units % 2 != 0 && tens % 2 != 0 && hundreds % 2 != 0);
9  }
10
11 int main()
12 {
13     int count = 0;
14     for (int num = 111; num <= 999; num++)
15     {
16         if (allDigitsOdd(num))
17         {
18             cout << num << endl;
19             count++;
20         }
21     }
22
23     cout << count << endl;
24     return 0;
25 }

```

Рисунок 2 — Текст программы (с использованием for)

Проект программы (с использованием while) изображен на рисунке 3, текст программы изображен на рисунке 4

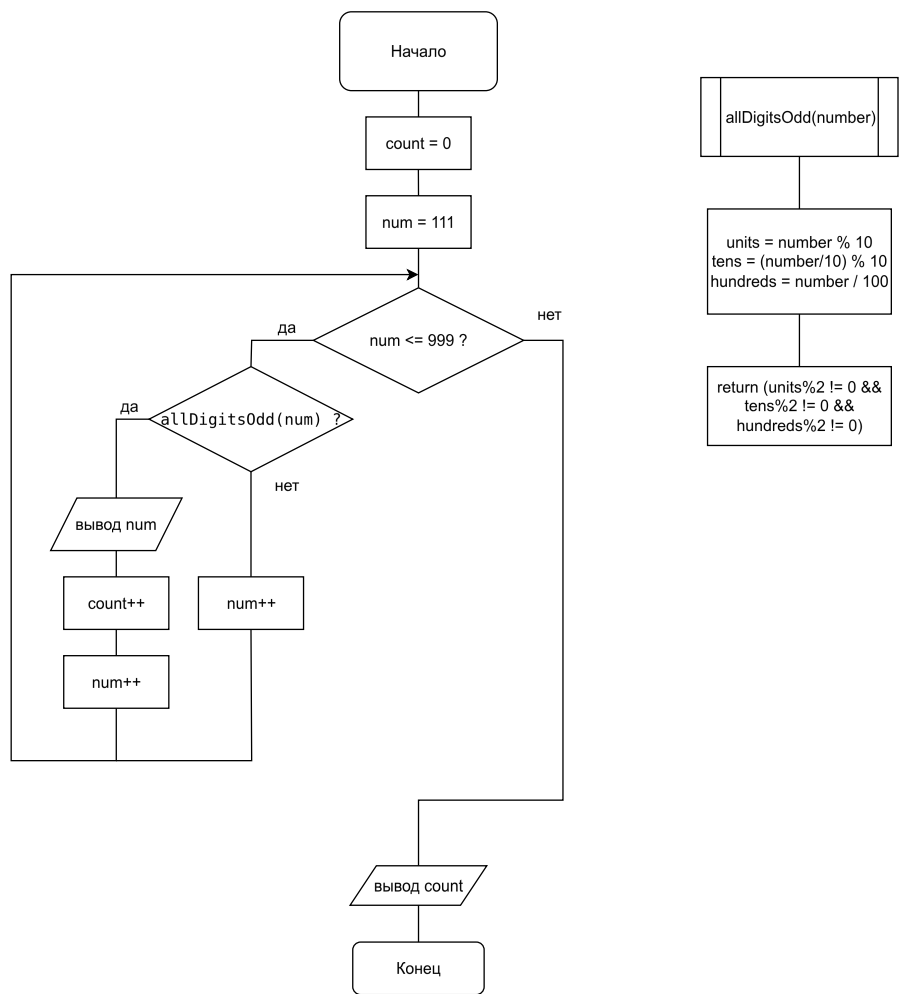


Рисунок 3 — Проект программы (с использованием while)

```

1  #include <iostream>
2  using namespace std;
3
4  bool allDigitsOdd(int number)
5  {
6      int units = number % 10;
7      int tens = (number / 10) % 10;
8      int hundreds = number / 100;
9      return (units % 2 != 0 && tens % 2 != 0 && hundreds % 2 != 0);
10 }
11
12 int main()
13 {
14     int num = 111;
15     int count = 0;
16
17     while (num <= 999)
18     {
19         if (allDigitsOdd(num))
20         { cout << num << endl;
21           count++;
22         }
23         num++;
24     }
25     cout << count << endl;
26     return 0;
27 }

```

Рисунок 4 — Текст программы (с использованием while)

Проект программы (с использованием do-while) изображен на рисунке 5, текст программы изображен на рисунке 6

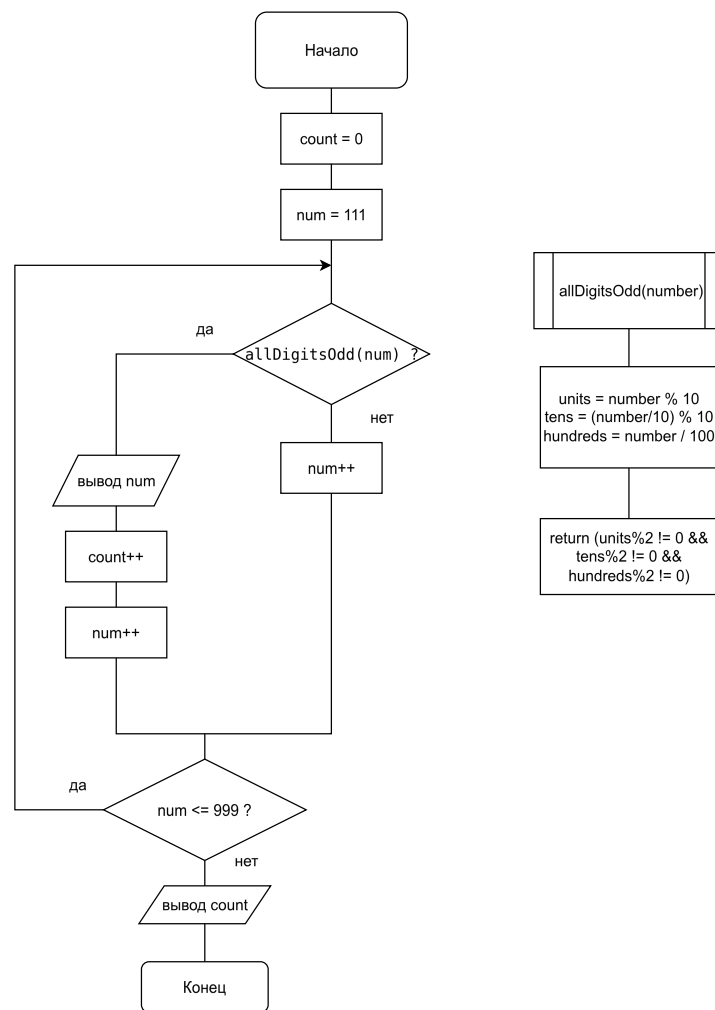


Рисунок 5 — Проект программы (с использованием do-while)

```

1  #include <iostream>
2  using namespace std;
3
4  bool allDigitsOdd(int number)
5  {
6      int units = number % 10;
7      int tens = (number / 10) % 10;
8      int hundreds = number / 100;
9      return (units % 2 != 0 && tens % 2 != 0 && hundreds % 2 != 0);
10 }
11
12 int main()
13 {
14     int num = 111;
15     int count = 0;
16
17     do
18     {
19         if (allDigitsOdd(num))
20         {
21             cout << num << endl;
22             count++;
23         }
24         num++;
25     } while (num <= 999);
26
27     cout << count << endl;
28     return 0;
29 }
  
```

Рисунок 6 — Текст программы (с использованием do-while)

Тестовые данные и результаты тестирования: тестовые данные и результат представлены в таблице 1

Таблица 1 — Тестовые данные и результат выполнения

цикл	Вывод программы
for	125
while	125
do-while	125

Вывод: в ходе данной лабораторной работы была успешно решена алгоритмическая задача нахождения всех натуральных чисел в диапазоне [111, 999], состоящих исключительно из нечетных цифр. Цель работы — освоение программирования циклических процессов — была достигнута путем практической реализации решения с использованием всех трех видов циклов: for, while, do-while.

Обоснование выбора наиболее рационального варианта: для решения данной задачи наиболее рациональным является цикл for.

Цикл for семантически предназначен для случаев, когда число итераций известно заранее или может быть легко вычислено, что полностью соответствует условию нашей задачи (диапазон от 111 до 999). Все управляющие элементы цикла (инициализация, условие, инкремент) сосредоточены в одной строке, что делает код структурированным и легко обозримым. Это снижает вероятность ошибок и упрощает его сопровождение.