МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

**Лабораторная работа № 2**

**«Реализация пошаговых блок-схем алгоритмов»**ОТЧЕТ

По лабораторной работе

по дисциплине

Информатика и компьютерные технологии

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Степаненко М.А.

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Субботин М.А.

24-ИВТ-4-1

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород 2025

**Цель работы:** Построить максимально подробную блок-схему алгоритма решения задачи.Разработать и визуализировать алгоритм для нахождения максимальной длины последовательности цифр, идущих подряд в произвольной строке, с помощью блок-схемы.

**Задачи:**

1. Формализовать логику решения задачи.
2. Построить блок-схему, отражающую все шаги алгоритма:
   * Ввод данных.
   * Посимвольный анализ строки.
   * Подсчет и сравнение длин последовательностей.
   * Вывод результата.
3. Убедиться, что блок-схема корректно обрабатывает крайние случаи.

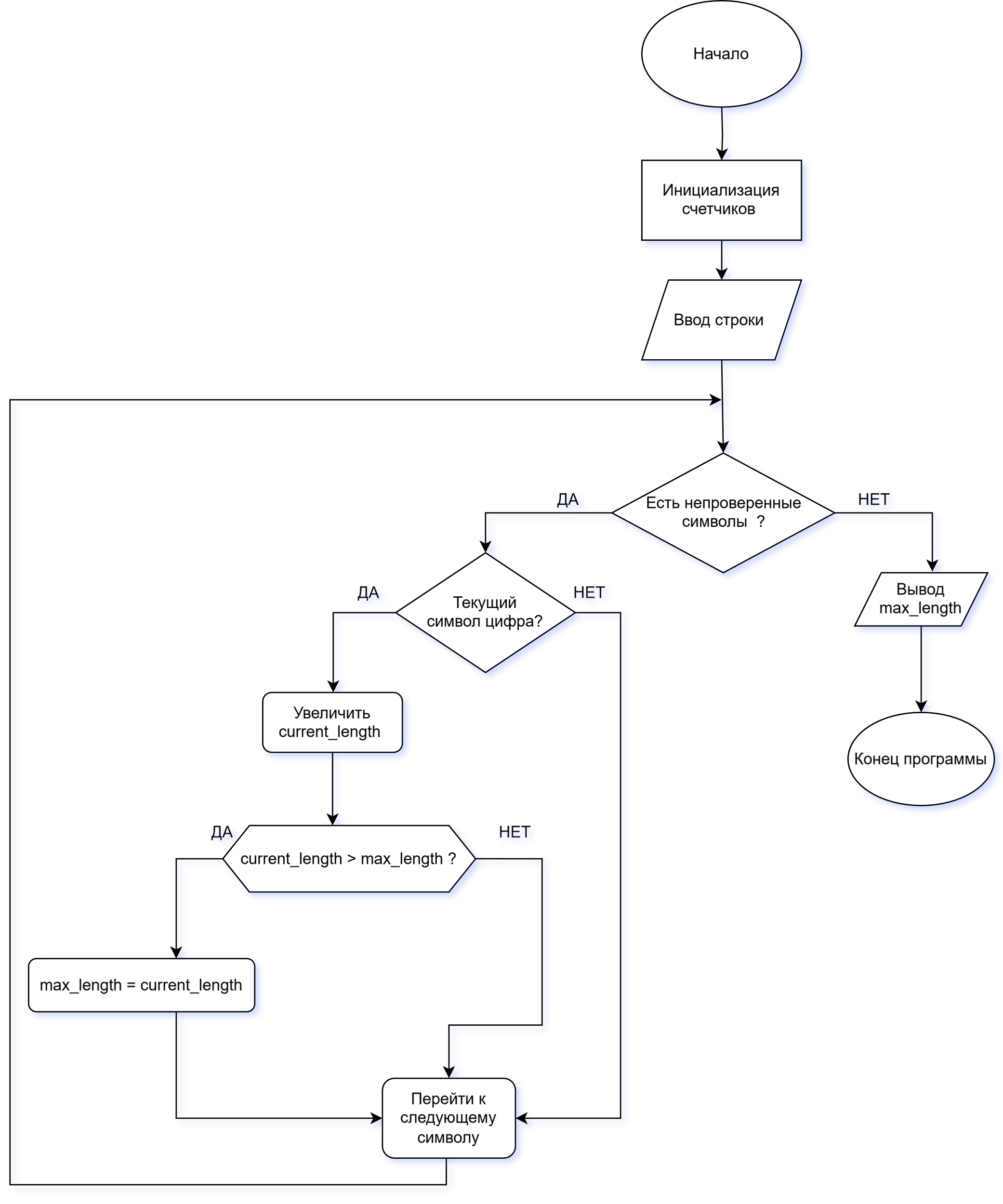
**Ход работы:**

1. **Старт алгоритма**
   * Начинаем с терминатора "Начало" - это отправная точка любой блок-схемы.
   * Из начального блока выходит единственная стрелка, указывающая на первый процесс.
2. **Инициализация переменных**
   * Третий блок - инициализация трех ключевых переменных:
3. max\_len = 0 (хранение максимальной длины последовательности)
4. current\_len = 0 (счетчик текущей последовательности)
5. i = 0 (индекс для перебора символов)
   * Этот блок объединяет все начальные установки в одном месте.
6. **Ввод строки** 
   * Следующий блок – ввод строки “S”.
   * Важно использовать параллелограмм для операций ввода/вывода.
   * Здесь пользователь предоставляет строку для анализа.
7. **Организация цикла**
   * Далее идет условие цикла в ромбе: i < длина строки? - проверка, остались ли символы для анализа.
   * Ромб имеет два выхода:
     1. Да - продолжаем обработку.
     2. Нет - завершаем алгоритм.
   * Это основное условие продолжения работы программы.
8. **Обработка символа**
   * Если символы еще есть, получаем текущий символ: c = S[i].
   * Следующее условие проверяет, является ли символ цифрой.
   * Это второе ключевое ветвление алгоритма.
9. **Обработка цифры**
   * Если символ цифра:
     1. Увеличиваем current\_len на 1.
     2. Проверяем, превышает ли current\_len max\_len.
     3. Если да - обновляем max\_len.
   * Эти действия идут последовательно друг за другом.
10. **Обработка не-цифры**
    * Если символ не цифра - сбрасываем current\_len в 0.
    * Это важно для корректного подсчета новых последовательностей.
11. **Переход к следующему символу**
    * После обработки любого символа увеличиваем индекс i на 1.
    * Возвращаемся к проверке условия цикла.
    * Так реализуется цикличность алгоритма.
12. **Вывод результата**
    * Когда все символы обработаны, выводим max\_len.
    * Используем параллелограмм для операции вывода.
    * Завершаем алгоритм терминатором “Конец”.

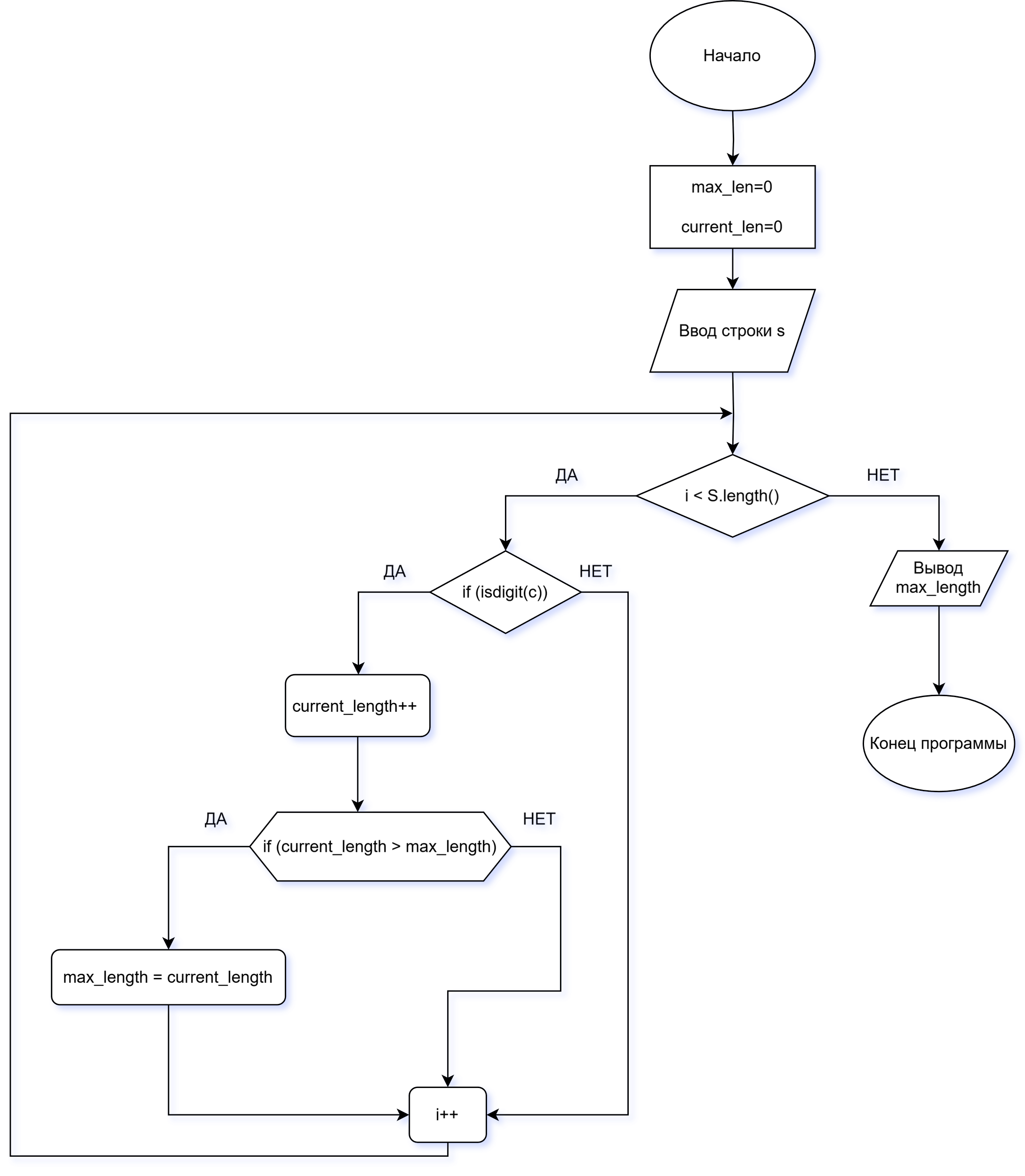
**Детализация логических связей**

1. **Основной цикл**
   * Цикл организован через возврат к условию "i < длина строки?"
   * Это классическая структура цикла while в блок-схемах
   * Обеспечивает последовательный перебор всех символов
2. **Вложенные условия**
   * Внутри цикла находится ветвление по типу символа
   * Каждая ветвь имеет свою последовательность действий
   * Все пути в конце сходятся на увеличении индекса
3. **Поток данных**
   * Переменные изменяются строго последовательно
   * max\_len обновляется только при выполнении условий
   * current\_len изменяется в двух различных ветвях

**Блок-схема алгоритма по заданию:**

****

**Блок-схема алгоритма по коду Си++:**

****

 **Код Си++ под блок-схему алгоритма:**

**Примеры:**

* "abc123def45" → ответ 3 (Последовательность "123")
* "no digits here"→ ответ 0 (Нет цифр)
* "1a22b333c4444" → ответ 4 (Последовательность “4444”)
* "" → ответ ( Пустая строка)

**Вывод:**

В ходе выполнения работы были освоены принципы алгоритмизации и визуализации вычислительных процессов при помощи блок-схем. На конкретном примере поиска максимальной последовательности цифр в строке отработаны ключевые этапы разработки алгоритма: от формализации задачи до создания наглядной блок-схемы, создания кода на языке Си++ включающей ввод данных, обработку строки, проверку условий и вывод результатов. Особое внимание уделено корректной обработке граничных случаев (отсутствие цифр, пустая строка) и оптимизации вычислительных ресурсов.

Приобретенные навыки позволяют эффективно проектировать алгоритмы для обработки строковых данных, что имеет практическое применение в анализе текстовой информации, валидации входных данных и обработке логов. Разработанная блок-схема может служить шаблоном для решения схожих задач, требующих последовательного анализа символов и поиска паттернов в строковых последовательностях. Полученный опыт подтверждает важность визуального представления алгоритмов на этапе их проектирования.