

Здесь будет титульник, листай ниже

# СОДЕРЖАНИЕ

|                                       |    |
|---------------------------------------|----|
| 1 ПОСТАНОВКА ЗАДАЧИ.....              | 5  |
| 1.1 Описание входных данных.....      | 5  |
| 1.2 Описание выходных данных.....     | 5  |
| 2 МЕТОД РЕШЕНИЯ.....                  | 6  |
| 3 ОПИСАНИЕ АЛГОРИТМОВ.....            | 7  |
| 3.1 Алгоритм функции main.....        | 7  |
| 4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....          | 8  |
| 5 КОД ПРОГРАММЫ.....                  | 9  |
| 5.1 Файл main.cpp.....                | 9  |
| 6 ТЕСТИРОВАНИЕ.....                   | 10 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ..... | 11 |

# **1 ПОСТАНОВКА ЗАДАЧИ**

Программа демонстрирует работу условного оператора и построение логического выражения. Написать программу, которая проверяет, принадлежит или нет введенный символ английскому алфавиту. Если принадлежит, то проверяет, является ли введенный символ гласным или согласным английского алфавита.

Использование массивов/строк/сторонних функций/т.д. для проверки не допускается

## **1.1 Описание входных данных**

## **1.2 Описание выходных данных**

## 2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- cin - объект стандартного потока ввода с клавиатуры;
- cout - объект стандартного потока вывода на экран;
- if..else - условный оператор.

## 3 ОПИСАНИЕ АЛГОРИТМОВ

Согласно этапам разработки, после определения необходимого инструментария в разделе «Метод», составляются подробные описания алгоритмов для методов классов и функций.

### 3.1 Алгоритм функции main

Функционал: основная функция программы.

Параметры: нет.

Возвращаемое значение: int - код ошибки.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции main

| № | Предикат  | Действия                          | № перехода |
|---|---|-----------------------------------|------------|
| 1 |   | объявление переменной n типа char | 2          |
| 2 |   | ввод значения переменной n        | 3          |
| 3 | ((int) n >= 65 && (int) n <= 90)    ((int) n >= 97 && (int) n <= 122)   |                                   | 4          |
|   |   | вывод на экран "Not letter"       | 5          |
| 4 | (int) n == 65    (int) n == 69    (int) n == 73    (int) n == 79    (int) n == 85    (int) n == 89    (int) n == 97    (int) n == 101    (int) n == 105    (int) n == 111    (int) n == 117    (int) n == 121 | вывод на экран "vowel"            | 5          |
|   |   | вывод на экран "consonant"        | 5          |
| 5 |   | возвращение 0                     | Ø          |

## 4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-1.

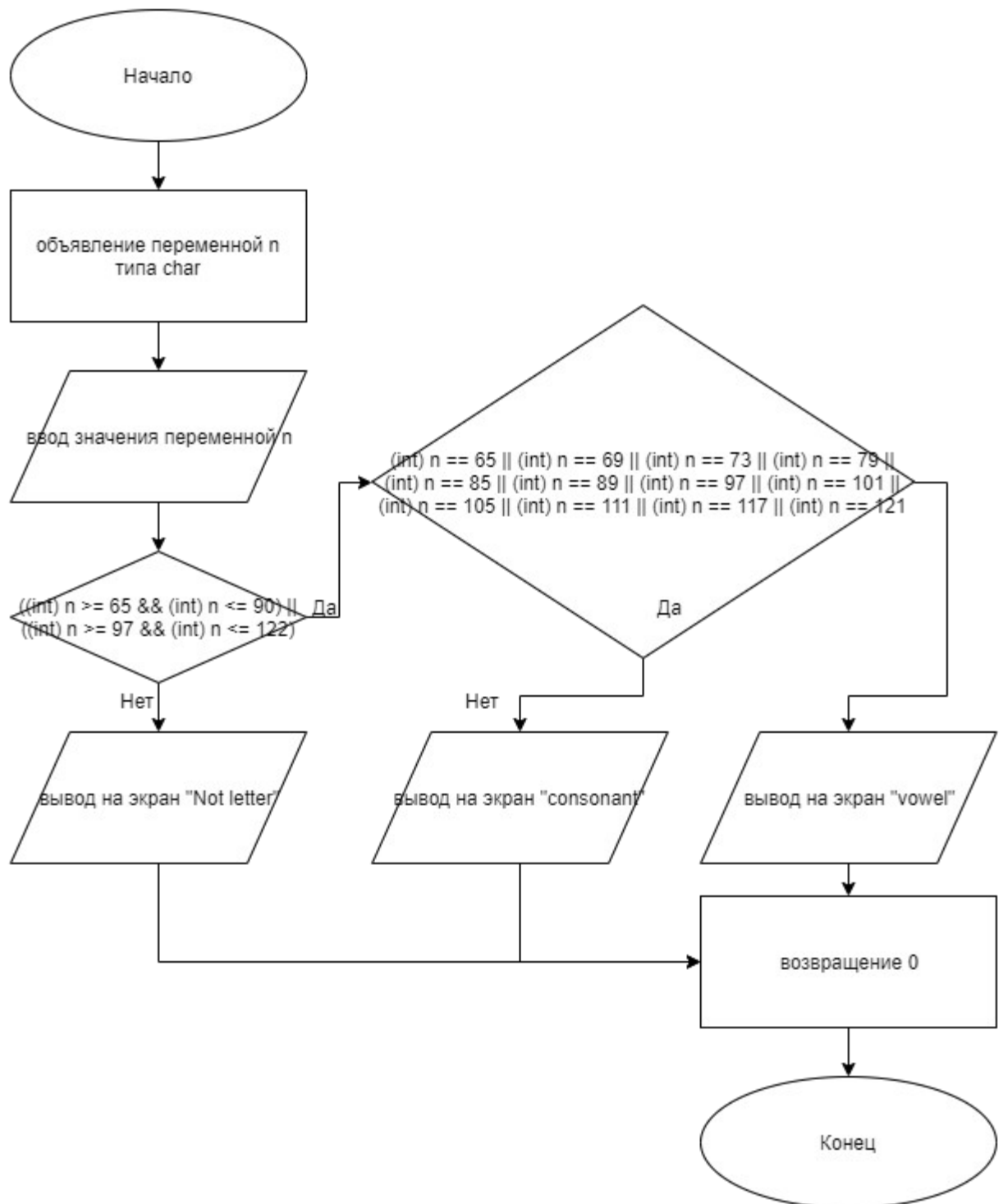


Рисунок 1 – Блок-схема алгоритма

## 5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

### 5.1 Файл main.cpp

*Листинг 1 – main.cpp*

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>

int main()
{
    char n;
    std::cin >> n;
    if (((int) n >= 65 && (int) n <= 90) || ((int) n >= 97 && (int) n <= 122))
    {
        if ((int) n == 65 || (int) n == 69 || (int) n == 73 || (int) n == 79 ||
            (int) n == 85 || (int) n == 89 || (int) n == 97 || (int) n == 101 || (int) n
            == 105 || (int) n == 111 || (int) n == 117 || (int) n == 121)
            std::cout << "vowel";
        else
            std::cout << "consonant";
    }
    else
        std::cout << "Not letter";
    return(0);
}
```

## 6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 2.

*Таблица 2 – Результат тестирования программы*

| <b>Входные данные</b> | <b>Ожидаемые выходные данные</b> | <b>Фактические выходные данные</b> |
|-----------------------|----------------------------------|------------------------------------|
| a                     | vowel                            | vowel                              |
| o                     | vowel                            | vowel                              |
| B                     | consonant                        | consonant                          |
| 1                     | Not letter                       | Not letter                         |



## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19 Единая система программной документации.
2. Методическое пособие студента для выполнения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] – URL: [https://mirea.aco-avvora.ru/student/files/methodichescoe\\_posobie\\_dlya\\_laboratornyh\\_rabot\\_3.pdf](https://mirea.aco-avvora.ru/student/files/methodichescoe_posobie_dlya_laboratornyh_rabot_3.pdf) (дата обращения 05.05.2021).
3. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: [https://mirea.aco-avvora.ru/student/files/Prilozheniye\\_k\\_methodichke.pdf](https://mirea.aco-avvora.ru/student/files/Prilozheniye_k_methodichke.pdf) (дата обращения 05.05.2021).
4. Шилдт Г. С++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2019. — 624 с.
5. Видео лекции по курсу «Объектно-ориентированное программирование» [Электронный ресурс]. АСО «Аврора».
6. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие /Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).