



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«МИРЭА - Российский технологический университет»

РТУ МИРЭА

Институт радиоэлектроники и информатики
Кафедра геоинформационных систем

ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 12
элементы алгоритмизации и процедурного программирования
«ИНФОРМАТИКА»

Выполнил студент группы АААА-00-00

Фамилия И.О.

Принял
Ассистент

Фамилия И.О.

Практическая
работа выполнена

«__» _____ 2023 г.

«Зачтено»

«__» _____ 2023 г.

Москва 2023

СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	3
1.1 Персональный вариант.....	3
2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ.....	4
2.1 Блок-схемы алгоритма программы.....	4
2.2 Код программы.....	11
2.3 Примеры тестирования программы.....	13
3 ВЫВОДЫ.....	15
4 ИНФОРМАЦИОННЫЙ ИСТОЧНИК.....	16

1 ПОСТАНОВКА ЗАДАЧИ

Требуется разработать блок-схему алгоритма и написать программу обработки данных. Требуется контролировать типы и диапазоны вводимых данных, предусмотреть обработку других исключительных ситуаций. Блок-схема должна описывать и процесс диалога с пользователем, и контроль вводимых данных, и подпрограммы вычислений с обработкой возможных исключительных операций. Блок-схема должна изображаться по ГОСТу. При обнаружении ошибки ввода или ошибки вычислений программа должна информативно уведомлять пользователя о причине ошибки. Если ошибка произошла на этапе ввода данных, программа должна просить пользователя повторить ввод.

1.1 Персональный вариант

2.17 Создать квадратную матрицу размера $M \times M$, где M является целым числом из диапазона $[2,5]$. Конкретный размер матрицы задается пользователем. Матрица должна содержать слова из 5ти букв английского алфавита, которые могут быть как случайными, так и вводиться пользователем. Полученную матрицу обработать следующим образом. Каждое слово матрицы заменить на другое слово, образованное из его букв, упорядоченных по алфавиту. Также требуется подсчитать общее количество гласных и согласных букв в исходной матрице. Результаты работы вывести на экран.

2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ

2.1 Блок-схемы алгоритма программы

Для удобства и повышения читаемости кода, основной алгоритм был разделен на функции. Для этого реализуем блок-схемы для каждой подпрограммы (рис. 1-6).

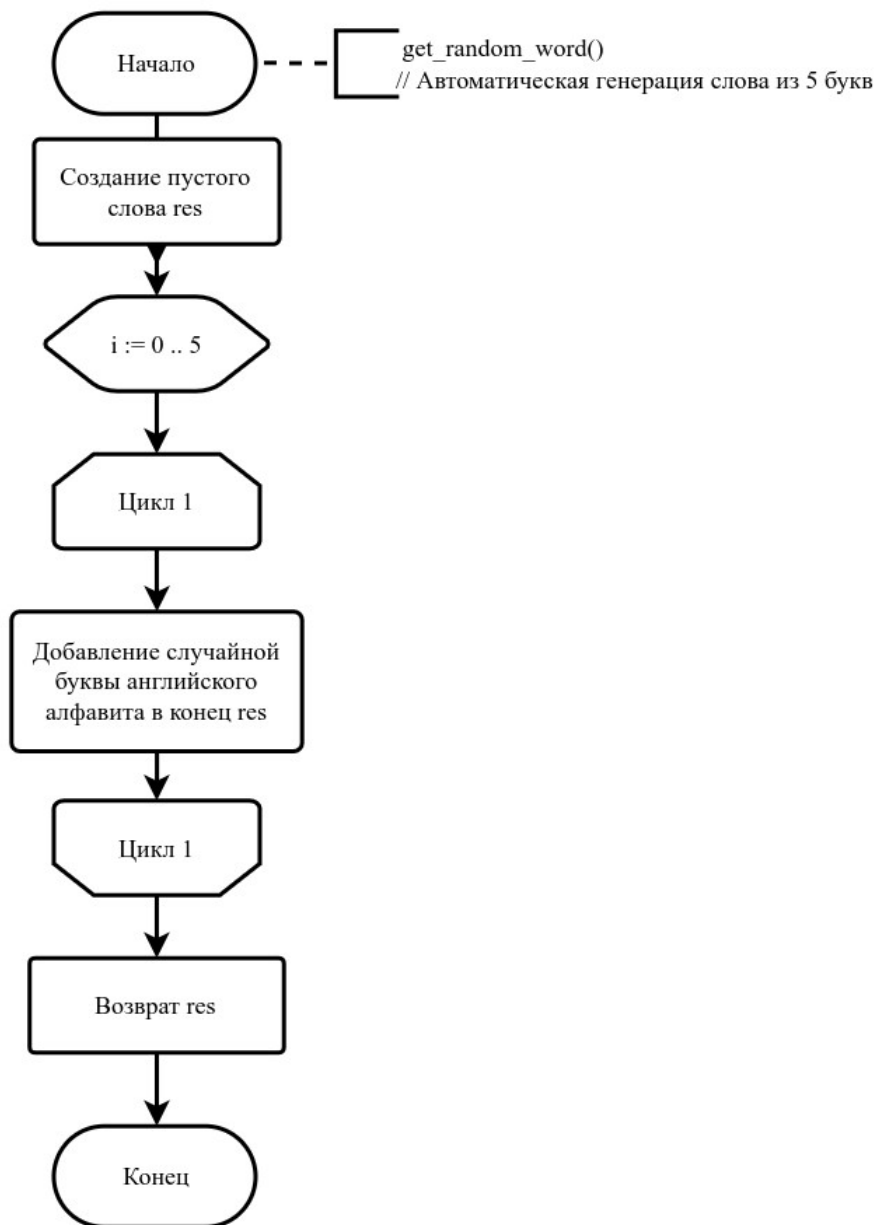


Рисунок 1 - Блок-схема подпрограммы `get_random_word`

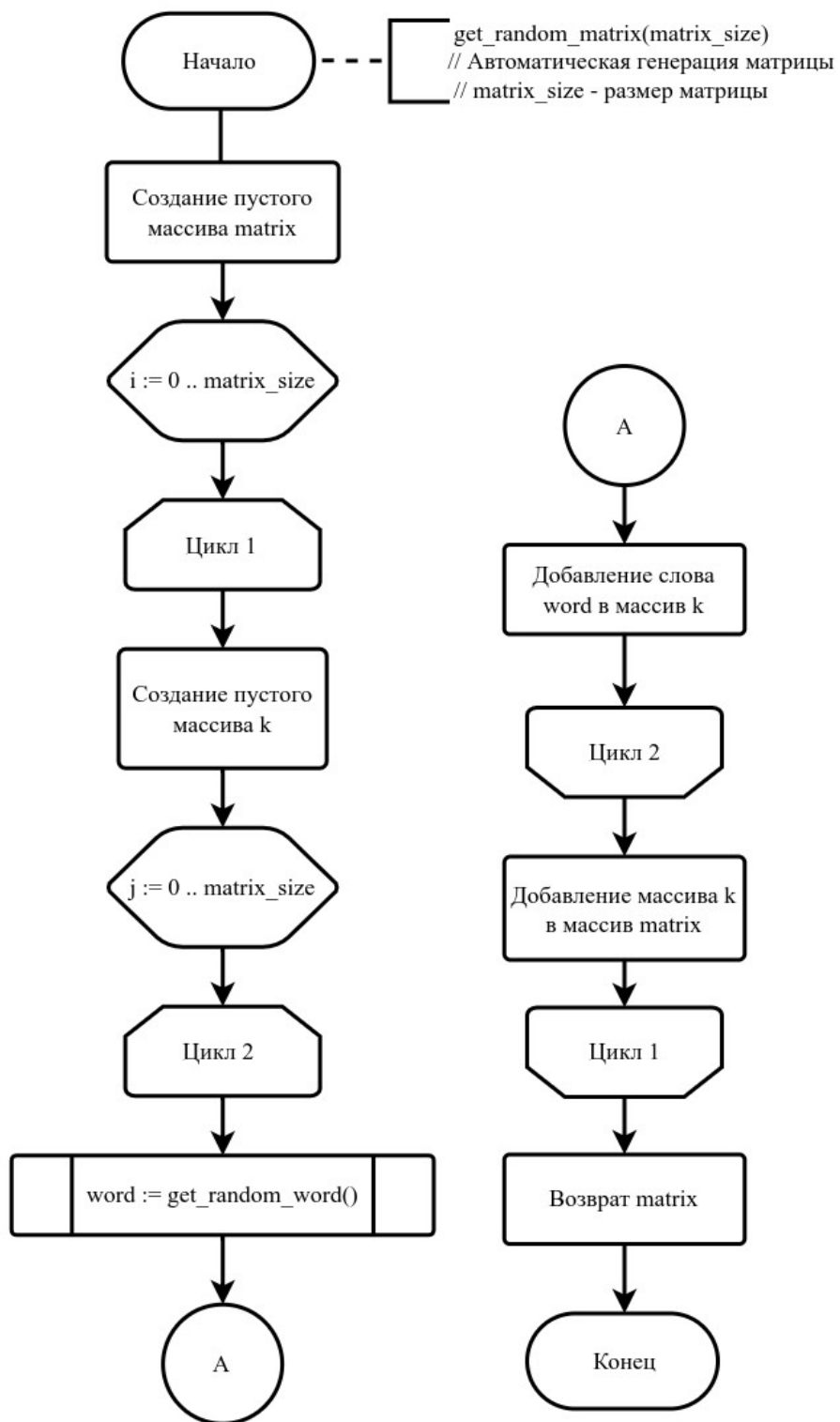


Рисунок 2 - Блок-схема подпрограммы `get_random_matrix`

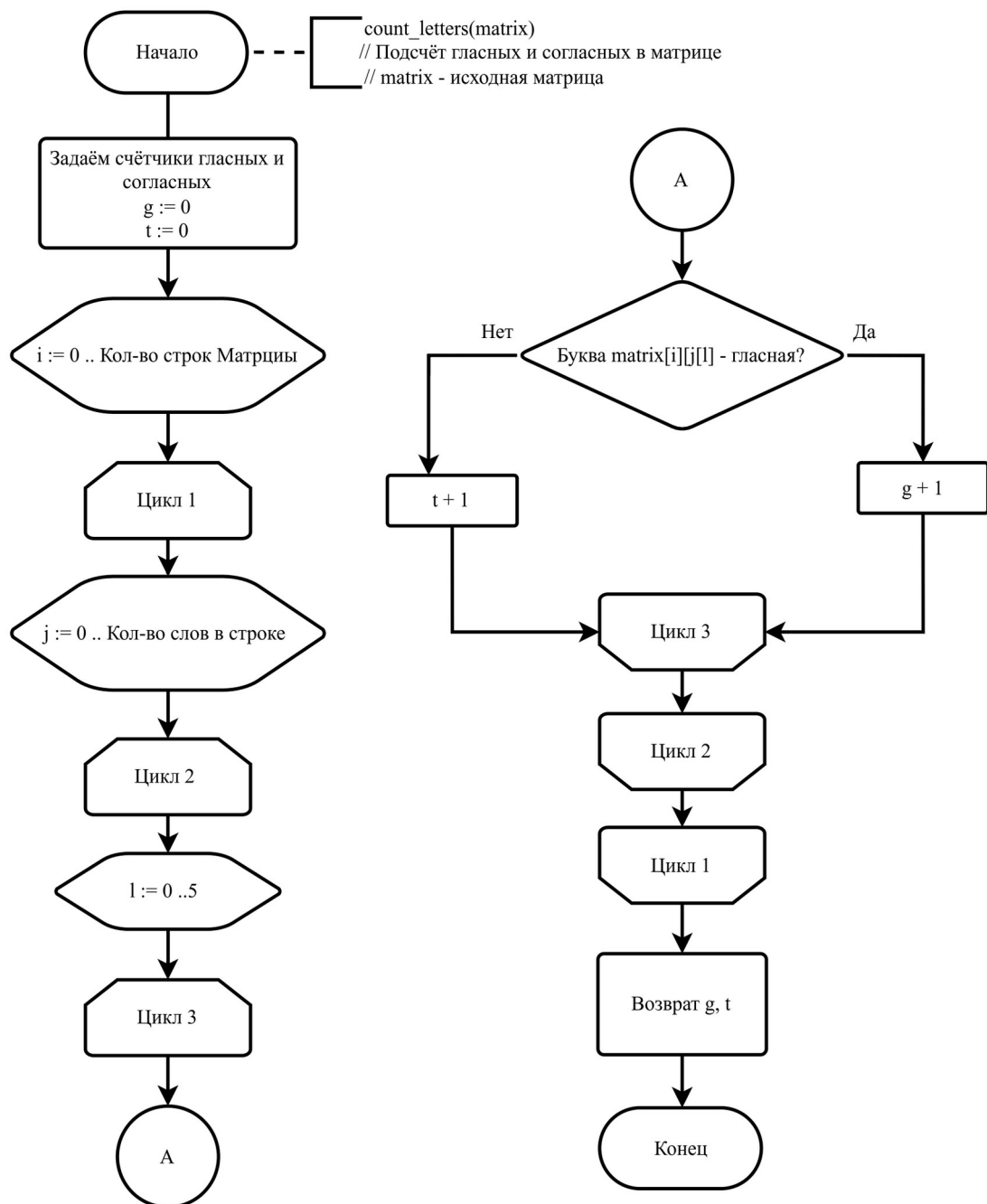


Рисунок 3 - Блок-схема подпрограммы count_letters

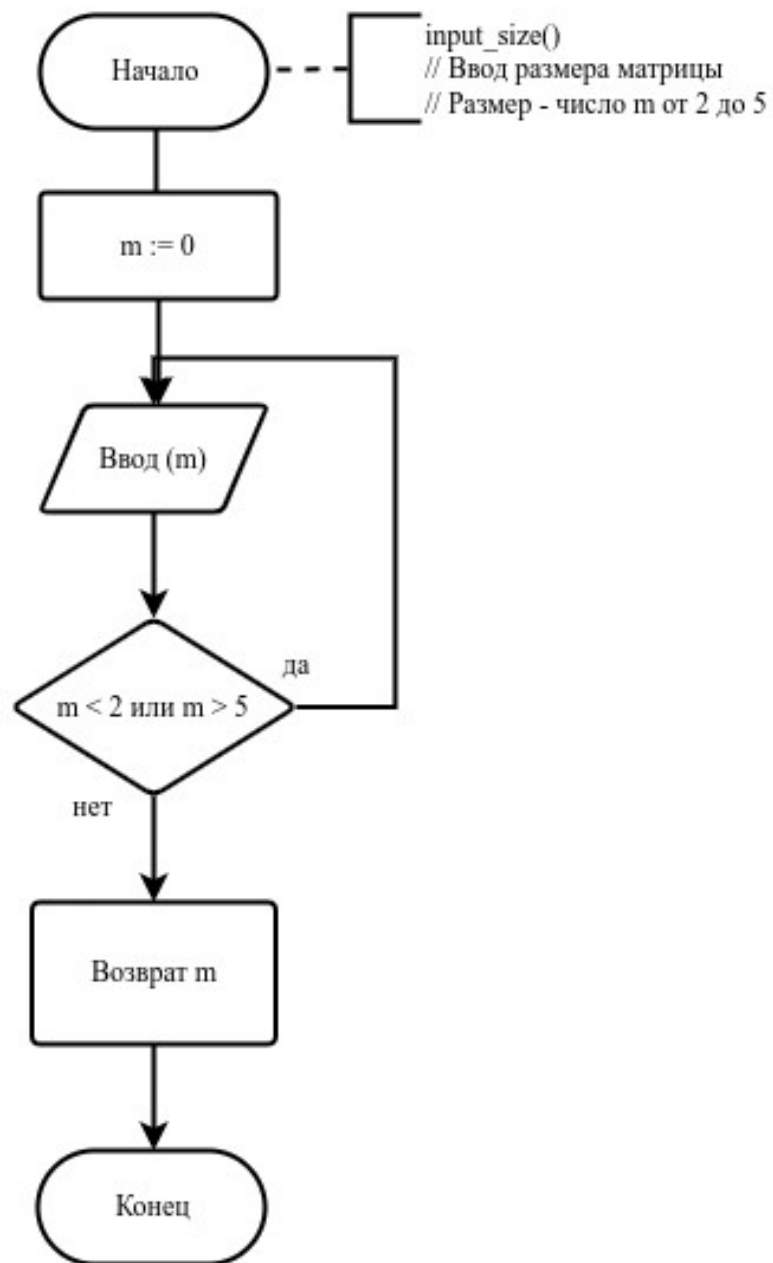


Рисунок 4 - Блок схема подпрограммы input_size()

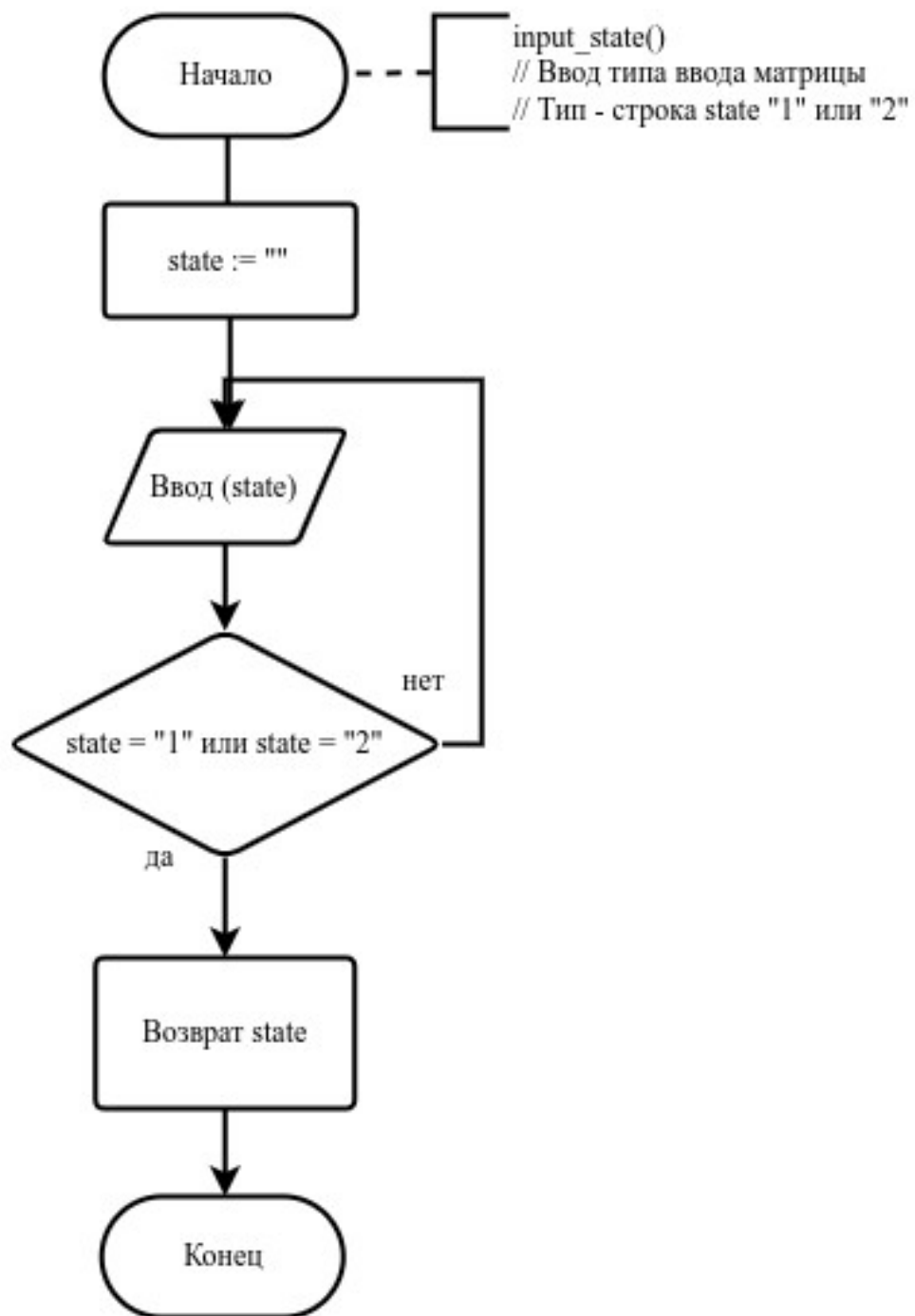


Рисунок 5 - Блок-схема подпрограммы input_state

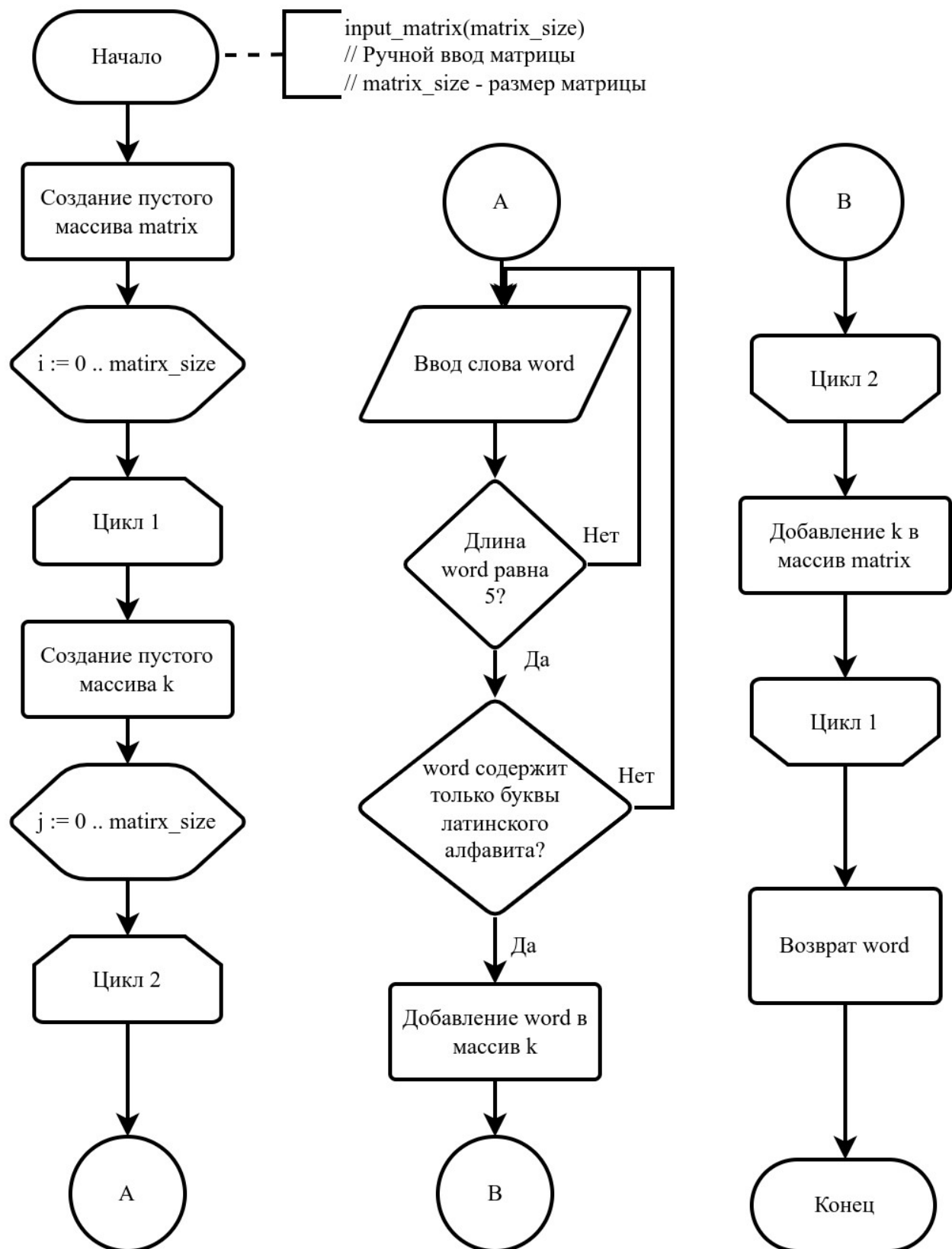


Рисунок 6 - Блок-схема подпрограммы input_matrix

Также блок-схема основной программы (рис. 7).

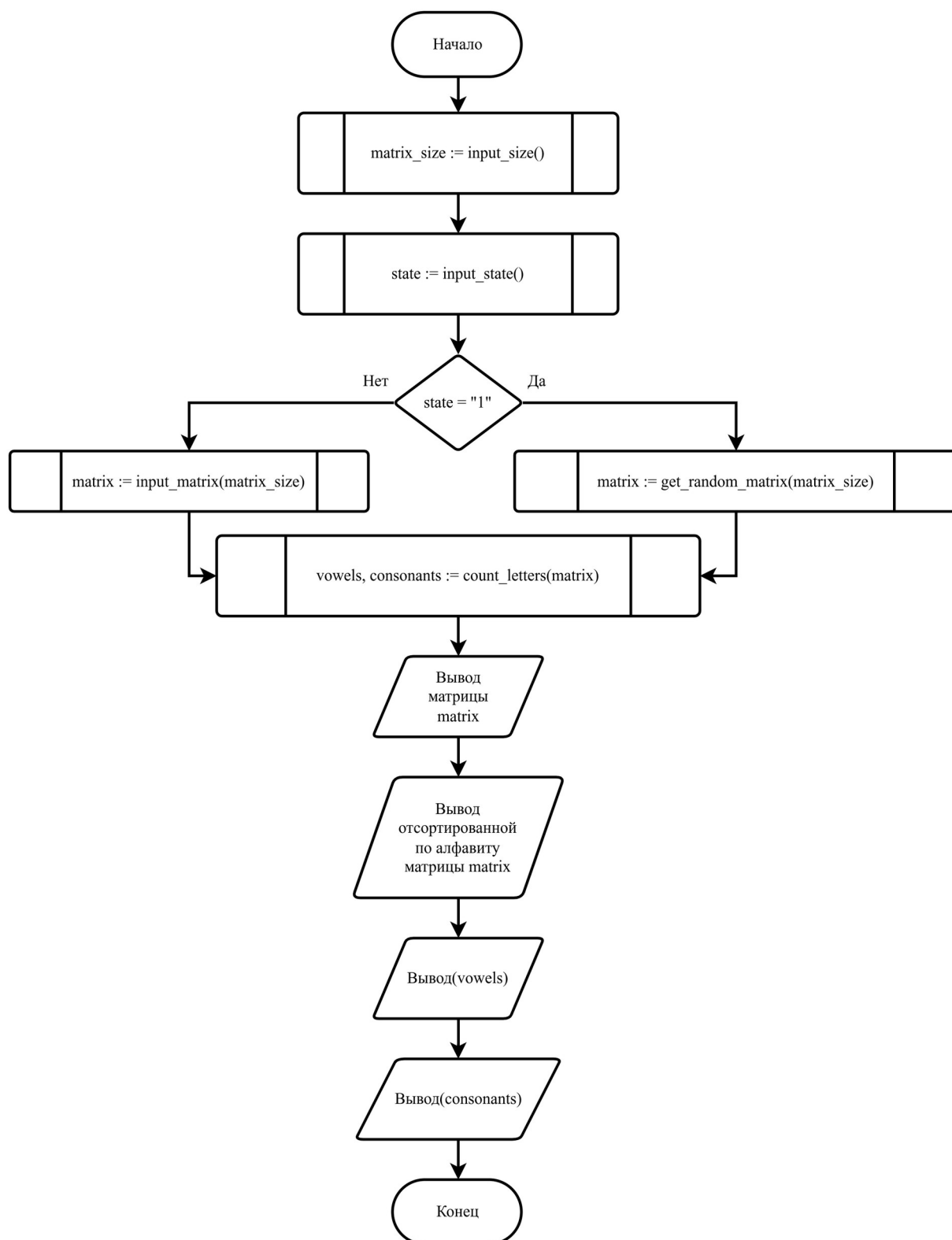


Рисунок 7 - Блок-схема основной программы

2.2 Код программы

Ниже приведены изображения кода программы (рис. 8, 9, 10).

```
# Функция получения слова из случайных символов латинского алфавита
def get_random_word():
    res = ""
    for i in range(5):
        # Выбираем случайную букву из алфавита и добавляем в конец результирующего слова
        res += ascii_lowercase[random.randint(a=1, b=25)]
    return res

# Функция для ввода значения размера матрицы
def input_size() -> int:
    m = ""
    while True:
        print("Введите число от 2 до 5 - размер матрицы:", end=" ")
        try:
            m = int(input().strip())
            if m < 2 or m > 5:
                print("Число должно быть в диапазоне от 2 до 5")
                continue
            break
        except:
            print("Некорректный ввод")
            continue
    return m

# Функция для ввода типа ввода матрицы
def input_state() -> str:
    state = ""
    while (True):
        print("Выберите метод заполнения матрицы (1 - автоматически, 2 - вручную):", end=" ")
        try:
            state = input().strip()
            if state == "1" or state == "2":
                break
            raise ValueError()
        except:
            print("Некорректный ввод")
            continue
    return state
```

Рисунок 8 - Подпрограммы get_random_word, input_size и input_state

```

# Функция для ввода матрицы вручную
def input_matrix(matrix_size: int):
    matrix = []
    print()
    print("Ввод слов длиной 5 в ячейки матрицы.")
    print("Слова должны состоять из букв латинского алфавита.")
    print()
    for i in range(matrix_size):
        k = []
        for j in range(matrix_size):
            while (True):
                print(f"Введите слово для ячейки матрицы для строки {i + 1} и столбца {j + 1}:", end=" ")
                try:
                    word = input().strip().lower()
                    if len(word) != 5:
                        print("Длина слова не равна 5.")
                        raise ValueError()
                    for letter in word:
                        if letter not in ascii_lowercase:
                            print("Слово содержит не только буквы латинского алфавита.")
                            raise ValueError()
                    k.append(word)
                    break
                except:
                    print("Некорректный ввод")
                    continue
            matrix.append(k)
    return matrix

# Функция для автоматического заполнения матрицы случайными словами
def get_random_matrix(matrix_size: int):
    matrix = []
    for i in range(matrix_size):
        k = []
        for j in range(matrix_size):
            word = get_random_word()
            k.append(word)
        matrix.append(k)

    return matrix

# Функция для подсчёта количества гласных и согласных в словах в матрице
def count_letters(matrix: list):
    vowels = "aeuyio" # Возможные гласные
    g = 0 # Счётчик гласных
    t = 0 # Счётчик согласных

    # Цикл по строкам матрицы
    for i in range(len(matrix)):
        # Цикл по словам строки
        for j in range(len(matrix[i])):
            # Цикл по буквам слова в строке
            for l in range(5):
                # Если буква гласная, то увеличиваем счётчик гласных, иначе согласных
                if matrix[i][j][l] in vowels:
                    g += 1
                else:
                    t += 1

    return g, t

```

Рисунок 9 - Подпрограммы input_matrix, get_random_matrix и count_letters

```

# Главная программа
if __name__ == '__main__':
    matrix_size = input_size() # Ввод размера матрицы
    state = input_state() # Ввод типа ввода матрицы

    matrix = []
    if state == "1":
        matrix = get_random_matrix(matrix_size)
    elif state == "2":
        matrix = input_matrix(matrix_size)

    vowels, consonants = count_letters(matrix)

    print("\nИсходная матрица:")
    for i in matrix:
        for j in i:
            print(j, end=" ")
        print()

    print("\nОтсортированная матрица:")
    for i in matrix:
        for j in i:
            print(''.join(sorted(j)), end=" ")
        print()

    print(f"\nГласных: {vowels}")
    print(f"Согласных: {consonants}")

```

Рисунок 10 - Главный алгоритм программы

2.3 Примеры тестирования программы

Ниже приведены примеры тестирования программы (рис. 11).

```

~/Документы/MIREA/12прак python3 main.py
Введите число от 2 до 5 - размер матрицы: 2
Выберите метод заполнения матрицы (1 - автоматически, 2 - вручную): 3
Некорректный ввод
Выберите метод заполнения матрицы (1 - автоматически, 2 - вручную): 2

Ввод слов длиной 5 в ячейки матрицы. Слова должны состоять из букв латинского алфавита.

Введите слово для ячейки матрицы для строки 1 и столбца 1: иб34п
Слово содержит не только буквы латинского алфавита.
Некорректный ввод
Введите слово для ячейки матрицы для строки 1 и столбца 1: gfnnf
Введите слово для ячейки матрицы для строки 1 и столбца 2: helpm
Введите слово для ячейки матрицы для строки 2 и столбца 1: eplea
Введите слово для ячейки матрицы для строки 2 и столбца 2: seiam

Исходная матрица:
gfnnf helpm
eplea seiam

Отсортированная матрица:
ffgnn ehlmp
aeelp aeims

Гласных: 7
Согласных: 13

~/Документы/MIREA/12прак python3 main.py
Введите число от 2 до 5 - размер матрицы: 4
Выберите метод заполнения матрицы (1 - автоматически, 2 - вручную): 1

Исходная матрица:
dcnzk geymb jqcbi vplom
rpmno jcjlj ngmoу wyjgc
hnfku wipcl uqikz esrsb
nfbks zjenq zspek lmwqm

Отсортированная матрица:
cdknz begny bcijq lmpov
mmopr cjjjl gmnoу cgjwy
fhknu cilpw ikquz berss
bfkns ejnqz ekpsz lmmqw

Гласных: 15
Согласных: 65

```

Рисунок 11 - Примеры тестирования программы

3 ВЫВОДЫ

В ходе практической работы была написана программа обработки данных, с учётом возможных ошибок при вводе, в соответствии с выбранным вариантом и разработана блок-схема алгоритма.

4 ИНФОРМАЦИОННЫЙ ИСТОЧНИК

Информатика : Методические указания по выполнению практических работ / С.С. Смирнов, Д.А. Карпов — М., МИРЭА — Российский технологический университет, 2020. – 102 с.