Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)

Институт радиоэлектроники и информационных технологий – РтФ

Школа бакалавриата

ОТЧЕТ

По дисциплине «Алгоритмизация и программирование»

Лабораторная работа №2

Студент: Шакертов Р.М.

Группа: РИЗ-111105у

Преподаватель: Саблина Н.Г.

Екатеринбург

2022

Содержание

1 Задача № 1 3

1.1 Описание условия задач 3

1.2 Постановка задачи 3

1.3 Метод решения задачи 3

1.3 Схема алгоритма 4

1.4 Исходный код 5

1.5 Результаты работы программы 5

2 Задача № 2 6

2.1 Описание условия задач 6

2.2 Постановка задачи 6

2.3 Метод решения задачи 6

2.4 Схема алгоритма 7

2.5 Исходный код 8

2.6 Результаты работы программы 9

1 Задача №1

1.1 Описание условия задачи

Заполнить символьный массив ТОР из 24 элементов с помощью датчика случайных чисел английскими заглавными буквами. Код буквы А равен 65. Вывести символьные значения элементов массива в строку через точку с запятой и пробел. Массив статический.

1.2 Постановка задачи

Исходные данные – натуральное число k, вводимое с клавиатуры пользователем.

Выходные данные – символьный массив tor, который выводится пользователю на дисплей.

1.3 Метод решения задачи

Программа получает от пользователя натуральное число k, которое является размерностью символьного массива tor.

Заглавной букве А английского алфавита соответствует код 65. В английском алфавите 26 букв – это значит, что заглавным буквам всего алфавита будет соответствовать диапазон чисел от 65 по 90. Для получения чисел из этого диапазона используется операция целочисленного деления и прибавляется сдвиг в 65 единиц относительно числовой оси. Соотношение выглядит следующим образом:

, (1)

где x – значение от 65 по 90,

rand() – функция случайных чисел.

1.4 Схема алгоритма

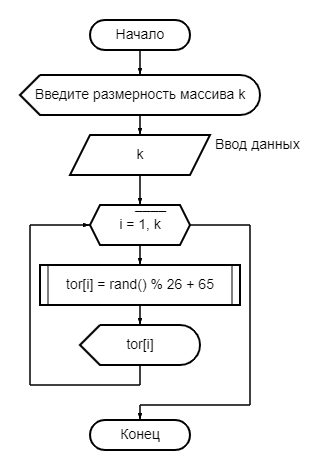


Рисунок 1 – Алгоритм вывода случайных заглавных букв англ. алфавита

1.5 Исходный код

#include <iostream>

**using** **namespace** std**;**

**int** main **()**

**{**

**char** tor**[**24**];**

**for(int** i **=** 0**;** i **<** 24**;** i**++)**

**{**

tor**[**i**]** **=** rand**()** **%** 26 **+** 65**;**

cout **<<** tor**[**i**]** **<<** "; "**;**

**}**

**}**

1.6 Результаты работы программы

Таблица 1 – Значения массива

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tor[1] | Tor[2] | Tor[3] | Tor[4] | Tor[5] | Tor[6] | Tor[7] | Tor[8] | Tor[9] | Tor[10] |
| 65 | 68 | 65 | 90 | 73 | 73 | 81 | 89 | 66 | 77 |
| Tor[11] | Tor[12] | Tor[13] | Tor[14] | Tor[15] | Tor[16] | Tor[17] | Tor[18] | Tor[19] | Tor[20] |
| 90 | 82 | 84 | 74 | 75 | 69 | 80 | 76 | 65 | 87 |
| Tor[21] | Tor[22] | Tor[23] | Tor[24] |  |  |  |  |  |  |
| 69 | 72 | 85 | 66 |  |  |  |  |  |  |

A; D; A; Z; I; I; Q; Y; B; M; Z; R; T; J; K; E; P; L; A; W; E; H; U; B;

2 Задача №2

2.1 Описание условия задачи

В заданной с клавиатуры матрице натуральных чисел МК размером A x B удалите строку с наибольшим количеством нулей. Если таких строк несколько, то удалить первую встретившуюся. Матрица статическая, размерность ее A = 5, B = 4.

2.2 Постановка задачи

Исходные данные – натуральное число n, a, b, вводимые пользователем с клавиатуры.

Выходные данные – массив натуральных чисел m, в котором удалена первая найденная строка с бо́льшим количеством нулей.

2.3 Метод решения задачи

Программа получает значения a и b натуральных чисел, которые определяют размерность массива, с которым взаимодействует программа – заполняет, перебирает и перезаписывает.

Основную задачу программа решает тремя циклами. С помощью первого цикла пользователь заносит построчно натуральные числа в массив – цикл заполнения. Затем второй цикл перебирает весь массив и ищет строку с максимальным количеством нулей для всего массива – цикл перебор. Третий цикл перезаписывает все строки, которые стоят ниже строки с большим количеством нулей, со сдвигом на эту строку.

2.4 Схема алгоритма

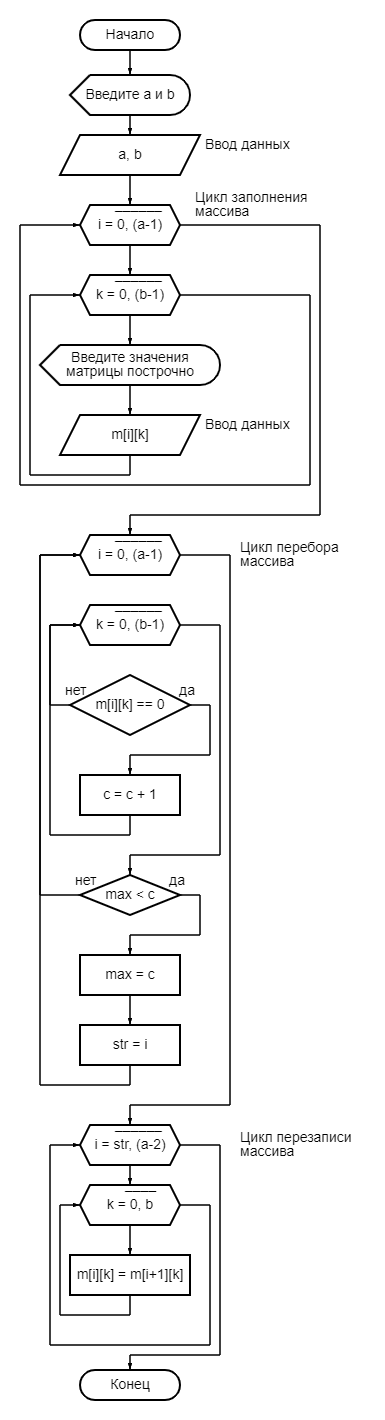


Рисунок 2 – Алгоритм удаления строки с наибольшим количеством нулей в массиве

2.5 Исходный код

#include <iostream>

**using** **namespace** std**;**

**int** main**()**

**{**​​​​​​​

**int** m**[**5**][**4**],** string**,** count**=**0**,** max**=**0**;**

**//Цикл заполнения**

**for(int** i **=** 0**;** i**<**5**;** i**++)**

**{**​​​​​​​

**for(int** z **=** 0**;** z **<** 4**;** z**++)**

**{**​​​​​​​

cout**<<**"Input actual value for matrix m[" **<<** i **<<** "][" **<<** z **<<** "]:" **<<** endl**;**

cin**>>**m**[**i**][**z**];**

**}**​​​​​​​

**}**​​​​​​​

**//Цикл перебора**

**for(int** i **=** 0**;** i **<** 5**;** i**++)**

**{**​​​​​​​

**for(int** z **=** 0**;** z **<** 4**;** z**++)**

**{**​​​​​​​

**if(**m**[**i**][**z**]** **==** 0**)**

**{**​​​​​​​

count **+=** 1**;**

**}**​​​​​​​

**}**​​​​​​​

**if(** max **<** count**)**

**{**​​​​​​​

max **=** count**;**

string **=** i**;**

**}**​​​​​​​

**}**​​​​​​​

**for(int** r **=** 0**;** r **<** 5 **;** r**++) //Цикл вывода исходного массива**

**{**​​​​​​​

**for(int** f **=** 0**;** f **<** 4**;** f**++)**

**{**​​​​​​​

**if(**f **%** 4 **==** 0**)** cout **<<** endl**;**

cout **<<** m**[**r**][**f**]** **<<** ";"**;**

**}**​​​​​​​

**}**​​​​​​​

cout **<<** endl**;**

**//Цикл удаления строки**

**for(int** k **=** string**;** k **<** 4**;** k**++)**

**{**​​​​​​​

**for(int** s **=** 0**;** s **<** 4**;** s**++)**

**{**​​​​​​​

m**[**k**][**s**]** **=** m**[**k**+**1**][**s**];**

**}**​​​​​​​

**}**​​​​​​​

**for(int** t **=** 0**;** t **<** 5**;** t**++)//Цикл вывода массива с удаленной строкой**

**{**​​​​​​​

**for(int** d **=** 0**;** d **<** 4**;** d**++)**

**{**​​​​​​​

**if(**d **%** 4 **==** 0**)** cout **<<** endl**;**

cout **<<** m**[**t**][**d**]** **<<** ";"**;**

**}**​​​​​​​

**}**​​​​​​​**}**​​​​​​​

2.6 Результат работы программы

Таблица 2 – Значения исходного массива

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | 0 | 1 | 2 | 3 |
| String 1 | 1 | 0 | 5 | 3 |
| String 2 | 5 | 0 | 0 | 7 |
| String 3 | 5 | 4 | 7 | 9 |
| String4 | 1 | 3 | 5 | 8 |
| String 5 | 8 | 8 | 3 | 1 |

Таблица 3 – Массив после удаления строки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | 0 | 1 | 2 | 3 |
| String 1 | 1 | 0 | 5 | 3 |
| String 2 | 5 | 4 | 7 | 9 |
| String 3 | 1 | 3 | 5 | 8 |
| String4 | 8 | 8 | 3 | 1 |
| String 5 | 8 | 8 | 3 | 1 |