Министерство образования и науки Российской Федерации

Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский**

**политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе №10**

Дисциплина: «Информатика»

Тема: Динамические массивы

Вариант 18

Выполнил:

Студент группы РИС-20-1б

Тараканов Д. М.

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

Пермь

2021 год

# **Цель работы**

Научится организации динамических массивов.

# **Постановка задачи**

Задача состоит в разработке программы, в которой создаётся динамический массив и выполнить обработку массива в соответствии со своим вариантом:

(18 вариант)

1. Сформировать массив строк.
2. Удалить из него строку, начинающуюся на букву “а”.

# **Анализ задачи**

1. Определить какие предстоит выполнить действия:

* Создать первый динамический массив строк.

string\* stroka = new string[size];

* Разработать функцию in для заполнения элементов динамического массива.

void out(string\*& arr, int N, int i = 0) {

while (i < N) {

cout << arr[i] << endl;

i++;

}

};

* Разработать функцию проверки первого символа строки в массиве, начинающегося на “a”.

bool Check(string row) {

bool otvet = false;

string check = {"Аа"};

string stroka = row;

for (int i = 0; i < size(check); i++) {

if (stroka[0] == check[i]) {

otvet = true;

}

}

return otvet;

};

* Реализовать удаление строки в массиве первый элемент которой равен “а” путём приравнивания строки к “0”.

for (i = 0; i < size; i++) {

if (Check(stroka[i]) == true) {

stroka[i] = "0";

counter++;

}

}

* Создать второй динамический массив в который войдут все строки из первого не равные “0”.

string\* row = new string[size - counter];

2. С какими типами данных действие надо сделать, в каком виде эти данные будут представлены:

* Для запоминания длины массива используем переменную size типа int.

int size=-1;

* Для подсчёта количества строк, начинающихся на “а”, используем переменную counter типа int.

int counter=0;

* Для обращения к первому динамическому массиву используем переменную указатель на первый элемент первого динамического массива типа string.

string\* stroka = new string[size];

* Для обращения ко второму динамическому массиву используем переменную указатель на первый элемент второго динамического массива типа string.

string\* row = new string[size - counter];

* Для функции in используем следующие элементы.

1. Массив строк типа string.
2. Длину динамического массива типа int.
3. Переменную отвечающую за продвижение по массиву строк типа int.

Сама функция имеет тип void так, как входе её работы функция не возвращает значений.

void in(string\*& arr, int N, int i = 0)

* Для функции out используем следующие элементы.

1. Массив строк типа string.
2. Длину динамического массива типа int.
3. Переменную отвечающую за продвижение по массиву строк типа int.

Сама функция имеет тип void так, как входе её работы функция не возвращает значений.

void out(string\*& arr, int N, int i = 0)

* Для функции Check используем следующие элементы.

1. Строку типа string.

Сама функция имеет тип bool так, как входе работы функции возвращается значение true, если первый элемент строки равен “а”.

bool Check(string row)

3. Для решения задачи данные были представлены в виде массива из слов (массив string)

4. Реализация операций ввода и вывода данных.

* Ввод данных с консоли реализован при помощи оператора cin.

while (i < N) {

cout << "Введиет строку: ";

cin >> arr[i];

i++;

}

* Вывод данных на консоль реализован при помощи оператора cout.

while (i < N) {

cout << arr[i] << endl;

i++;

}

5. Поставленные задачи будут решены при помощи следующих действий:

* Программа запрашивает у пользователя длину первого динамического массива.

while (size < 1) {

cout << "\nВведите кол-во строк: ";

cin >> size;

if (size < 1) {

cout << "Недопустимое кол-во строк!";

}

}

* При помощи функции in происходит заполнение элементов первого динамического массива.

void in(string\*& arr, int N, int i = 0)

{

while (i < N) {

cout << "Введиет строку: ";

cin >> arr[i];

i++;

}

};

* При помощи оператора for и функции Check проверяется первый элемент каждой строки первого массива, и если проверка возвращает true, то программа заменяет строку на “0” и увеличивает переменную counter на 1.

for (i = 0; i < size; i++) {

if (Check(stroka[i]) == true) {

stroka[i] = "0";

counter++;

}

}

bool Check(string row) {

bool otvet = false;

string check = { "Аа" };

string stroka = row;

for (int i = 0; i < size(check); i++) {

if (stroka[0] == check[i]) {

otvet = true;

}

}

return otvet;

};

* Используя оператор if в программе рассмотрена возможность совпадения size и сounter. В таком случае программа выведет сообщение на консоль о том, что все строки начинаются на “а”.

if (counter == size) {

cout << "\nВсе строки в массиве начинаются на 'а'!\n";

}

* Используя оператор if в программе рассмотрена возможность того что counter будет равен 0. В таком случае программа выведет сообщение на консоль о том, что в массиве нет строк, начинающихся на “а”.

else if (counter == 0) {

cout << "\nВ массиве нет строк начинающихся на 'а'!\n";

}

* При помощи оператора for программа дублирует элементы не равные “0” из первого массива во второй массив.

for (i = 0; i < size; i++) {

if (stroka[i] != "0") {

row[j] = stroka[i];

j++;

}

}

* Программа производит удаление первого динамического массива.

delete[] stroka;

* При помощи функции out программа выводит элементы второго динамического массива.

void out(string\*& arr, int N, int i = 0) {

while (i < N) {

cout << arr[i] << endl;

i++;

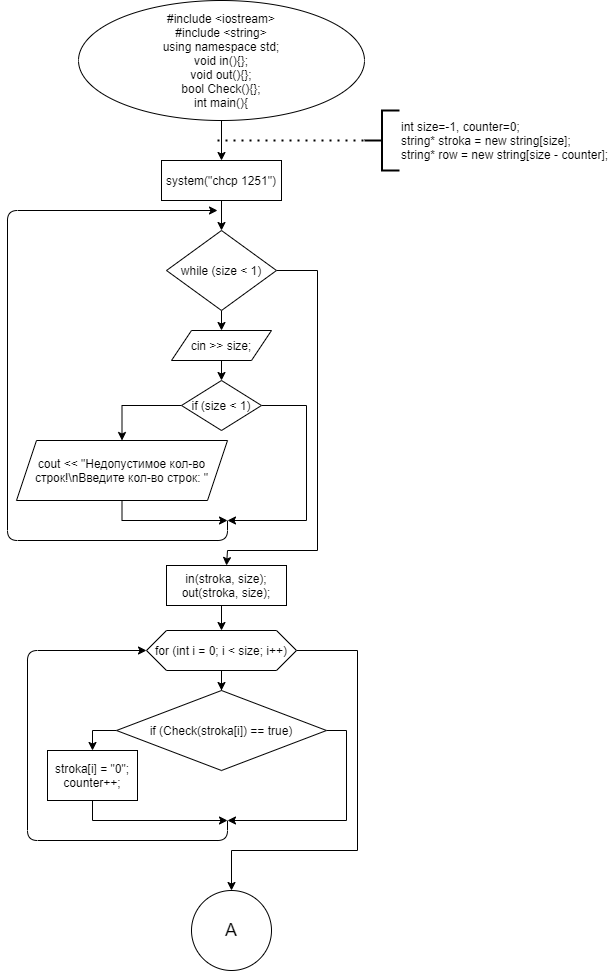
}

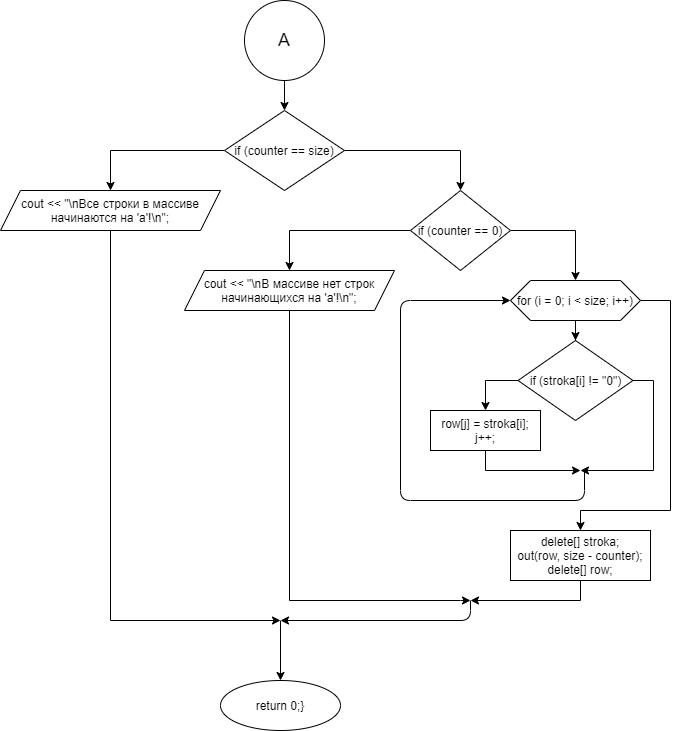
};

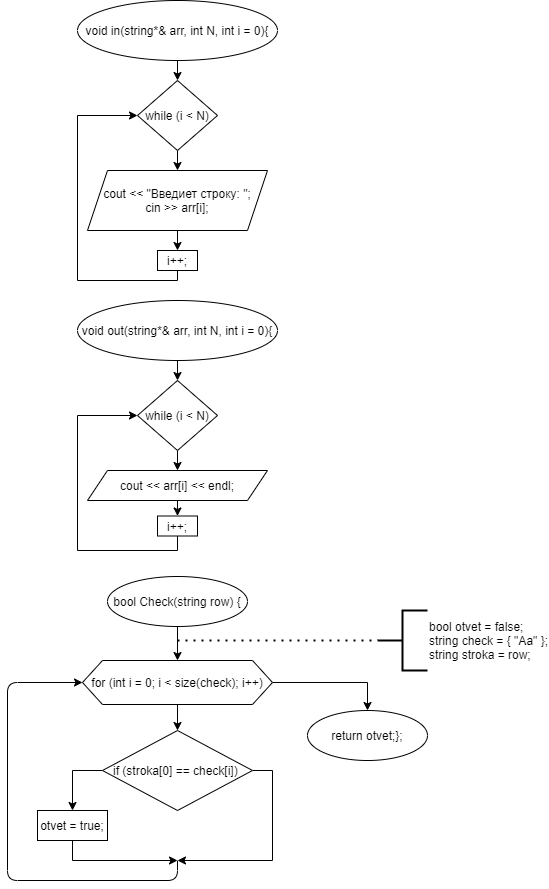
* Программа производит удаление второго динамического массива.

delete[] row;

# **Блок схема**







# **Код программы на языке C++**

#include <iostream>

#include <string>

using namespace std;

//Функция ввода элементов в динамический массив

void in(string\*& arr, int N, int i = 0)

{

while (i < N) {

cout << "Введиет строку: ";

cin >> arr[i];

i++;

}

};

//Функция вывода элемнтов в динамического массива

void out(string\*& arr, int N, int i = 0) {

while (i < N) {

cout << arr[i] << endl;

i++;

}

};

//Функция проверки первого элемента массива

bool Check (string row) {

bool otvet=false;

string check = {"АаAa"};

string stroka = row;

for (int i = 0; i < size(check); i++) {

if (stroka[0] == check[i]) {

otvet = true;

}

}

return otvet;

}

int main()

{

system("chcp 1251");

int size=-1,counter=0, i, j=0;

while (size < 1) {

cout << "\nВведите кол-во строк: ";

cin >> size;

if (size < 1) {

cout << "Недопустимое кол-во строк!";

}

}

string\* stroka = new string[size];

in(stroka, size);

cout << "\nМассив строк:\n";

out(stroka, size);

for (i = 0; i < size; i++) {

if (CheckAndCount(stroka[i]) == true) {

stroka[i] = "0";

counter++;

}

}

if (counter == size) {

cout << "\nВсе строки в массиве начинаются на 'а'!\n";

}

else if (counter == 0) {

cout << "\nВ массиве нет строк начинающихся на 'а'!\n";

}

else {

string\* row = new string[size - counter];

for (i = 0; i < size; i++) {

if (stroka[i] != "0") {

row[j] = stroka[i];

j++;

}

}

delete[] stroka;

cout << "\nКол-во удалённых строк = "<< counter <<"\nМассив строк после удаления:\n";

out(row, size - counter);

delete[] row;

} system("pause");

return 0;

}

# **Скриншоты тестов**

