Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение

высшего образования

«**Пермский национальный исследовательский**

**политехнический университет»**

Кафедра «Информационные технологии и автоматизированные

системы»

**ОТЧЁТ**

**по лабораторной работе №10**

Дисциплина: «Информатика»

Тема: Динамические массивы

Вариант 17

Выполнил:

Студент группы ИВТ-20-2б

Сафронов Владислав Владиславович

Проверила:

Доцент кафедры ИТАС

Полякова О.А.

**Пермь, 2021**

**Цель задачи**

Организация динамических массивов.

**Постановка задачи**

Сформировать массив строк. Удалить из него самую длинную строку. **Анализ задачи**

**Какие типы данных будут использованы:**

* Символьный тип данных char[] используется для хранения и генерации строк.
* Целочисленный тип данных integer используется для хранения и ввода размера массива, индексов элементов, параметра в циклах.

**Какие с этими данными выполняются действия:**

* char[] – добавление строки в массив;
* integer – ввод с клавиатуры, инкремент, присваивание.

**В каком виде эти данные будут представлены:**

1. Функция make\_array для создания массива строк. В качестве параметра указывается размерность массива. Строки генерируются с помощью генератора случайных чисел.

char\*\* make\_array(int size)

{

char\*\* a;

a = new char\* [size];

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

{

int len = rand() % 30 + 1;

a[i] = new char[len + 1];

for (int j = 0; j < len; j++)

{

a[i][j] = rand() % 57 + 65;

}

a[i][len] = '\0';

}

return a;

}

1. Функция print\_array для вывода массива строк в консоль. В качестве входных параметров указывается массив указателей на char[](массив строк) и размерность массива.

void print\_array(char\*\* a, int size)

{

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

{

cout << '\t';

int k = 0;

while (a[i][k])

{

cout << a[i][k];

k++;

}

cout << '\n';

}

}

1. Функция find\_and\_delete\_max\_string, которая ищет в массиве самую длинную строку и удаляет её. В качестве входных параметров указывается массив указателей на char[](массив строк) и ссылка на переменную типа int, в которой хранится размер массива.

void find\_and\_delete\_max\_string(char\*\*& a, int& size)

{

int i;

int max\_i = 0;

int max\_k = 0;

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

{

int k = 0;

while (a[i][k])

{

k++;

}

if (k > max\_k)

{

max\_k = k;

max\_i = i;

}

}

delete[] a[max\_i];

size--;

char\*\* new\_a = new char\* [size];

for (i = 0; i < max\_i; i++)

{

new\_a[i] = a[i];

}

for (i; i < size; i++)

{

new\_a[i] = a[i + 1];

}

delete[] a;

a = new\_a;

new\_a = nullptr;

}

**Какими операторами будет организован ввод и вывод:**

* Ввод данных осуществляется через консоль с помощью функции cin;
* Вывод данных в консоль осуществляется с помощью функции cout.

**Описание функции main():**

Подключение русского языка:

setlocale(LC\_ALL, "Russian");

Функция srand для работы с генератором случайных чисел:

srand(time(NULL));

Резервирование памяти под переменные и ввод размерности массива:

int n;

char\*\* arr;

cout << "Введите количество строк n: "; cin >> n;

while (n < 3) {

cout << "n должно быть больше или равно 3.";

cout << "\nВведите количество строк n: "; cin >> n;

}

Генерация массива строк с помощью функции make\_array и вывод сгенерированного массива в консоль с помощью функции print\_array:

cout << "\nГенерация массива из " << n << " случайно сгенерированных строк...";

arr = make\_array(n);

cout << "\nПолученный массив: \n";

print\_array(arr, n);

cout << "\n";

Поиск и удаление самой длинной строки из массива с помощью функции find\_and\_delete\_max\_string с последующим выводом нового массива в консоль:

cout << "\nУдаление самой длинной строки из массива...\n";

find\_and\_delete\_max\_string(arr, n);

cout << "\nПолученный массив: \n";

print\_array(arr, n);

Освобождение памяти:

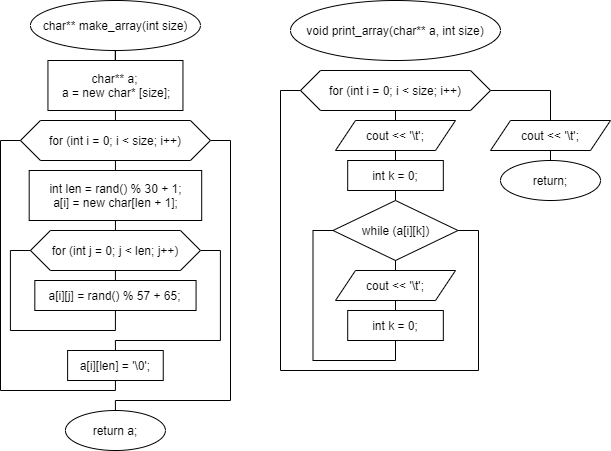
for (int i = 0; i < n; i++) {

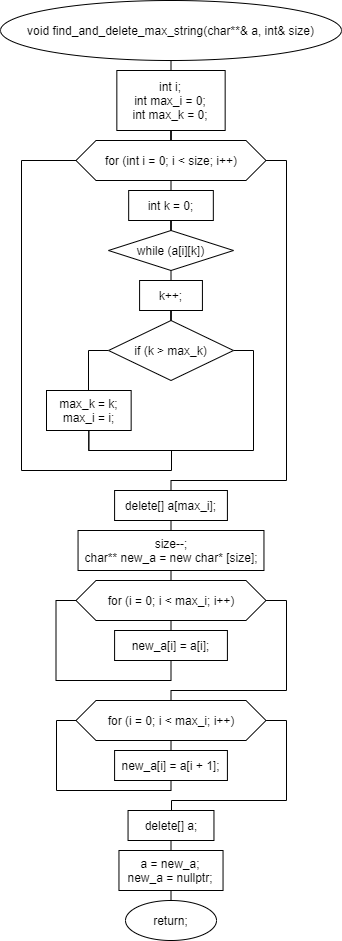
delete[] arr[i];

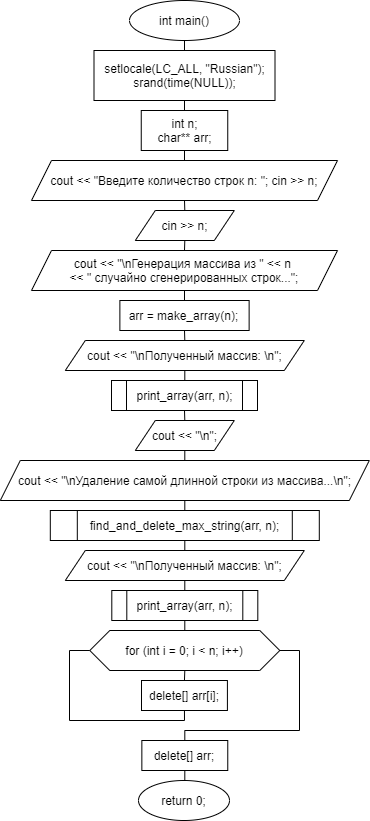
}

delete[] arr;

**Блок-схема программы**







**Код программы**

#include <iostream>

#include <ctime>

using namespace std;

char\*\* make\_array(int size)

{

char\*\* a;

a = new char\* [size];

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

{

int len = rand() % 30 + 1;

a[i] = new char[len + 1];

for (int j = 0; j < len; j++)

{

a[i][j] = rand() % 57 + 65;

}

a[i][len] = '\0';

}

return a;

}

void print\_array(char\*\* a, int size)

{

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

{

cout << '\t';

int k = 0;

while (a[i][k])

{

cout << a[i][k];

k++;

}

cout << '\n';

}

}

void find\_and\_delete\_max\_string(char\*\*& a, int& size)

{

int i;

int max\_i = 0;

int max\_k = 0;

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

{

int k = 0;

while (a[i][k])

{

k++;

}

if (k > max\_k)

{

max\_k = k;

max\_i = i;

}

}

delete[] a[max\_i];

size--;

char\*\* new\_a = new char\* [size];

for (i = 0; i < max\_i; i++)

{

new\_a[i] = a[i];

}

for (i; i < size; i++)

{

new\_a[i] = a[i + 1];

}

delete[] a;

a = new\_a;

new\_a = nullptr;

}

int main()

{

setlocale(LC\_ALL, "Russian");

srand(time(NULL));

int n;

char\*\* arr;

cout << "Введите количество строк n: "; cin >> n;

while (n < 3) {

cout << "n должно быть больше или равно 3.";

cout << "\nВведите количество строк n: "; cin >> n;

}

cout << "\nГенерация массива из " << n << " случайно сгенерированных строк...";

arr = make\_array(n);

cout << "\nПолученный массив: \n";

print\_array(arr, n);

cout << "\n";

cout << "\nУдаление самой длинной строки из массива...\n";

find\_and\_delete\_max\_string(arr, n);

cout << "\nПолученный массив: \n";

print\_array(arr, n);

for (int i = 0; i < n; i++)

{

delete[] arr[i];

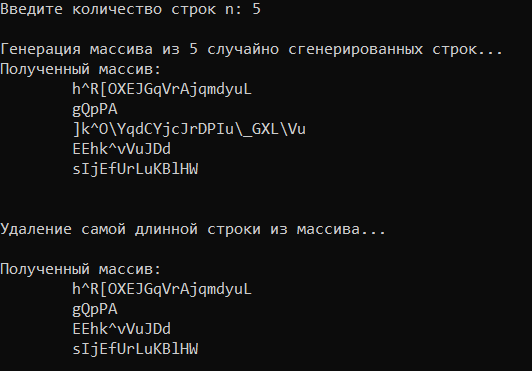
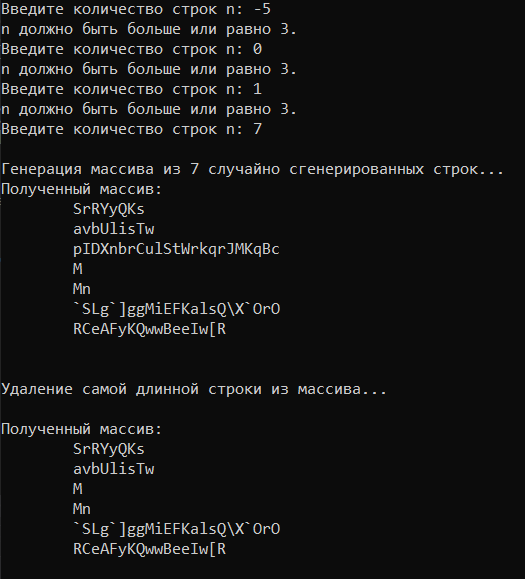
}

delete[] arr;

return 0;

}

**Скриншоты результатов работы программы**

****