МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

МОСКОВСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

Институт информационных технологий (ИТ)

Кафедра математического обеспечения и стандартизации информационных

технологий (МОСИТ)

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 2**

**по дисциплине «***Структуры и алгоритмы обработки данных***»**

**тема «**Структура данных массив. Указатели и массивы. Функции**»**

Выполнил: студент группы ИВБО-02-15 Прохоров А.В.

Приняла: старший преподаватель кафедры Скворцова Л.А.

**1. Условие задания**

Цель. Приобретение навыков по выполнению операций над массивом в языке С++.

Получение навыков по разработке функций

**Задание 1.** Разработайте приложение для выполнения операций со статическим массивом.

1. Разработать функции для заполнения и отображения значений массива.
2. Разработать функции для выполнения всех задач варианта. Выполнить декомпозицию задач варианта и оформите подзадачи в виде отдельных функций.
3. Разработать программу. Программа должна содержать меню, позволяющее продемонстрировать работу всех операций над массивом.
4. Отладить программу. Привести в конце модуля программы тесты.

**Задание 2.** Разработайте программу для выполнения всех операций варианта над динамическим массивом.

**Задание 3.** Подготовьте отчет по выполненной работе по статическому и динамическому массивам. Ответьте на вопросы данной лабораторной работы. Включите в отчет описание алгоритма сортировки на естественном языке. Пример оформления отчета приведен в конце данного документа.

**Вариант 20 (3):**

1. Вставить новое значение в позицию каждого элемента, которое содержит число, являющееся степенью двух.
2. Найти последнее вхождение в массив числа, у которого равны первая и последняя цифры.
3. Расположить все числа массива, которые больше первого числа в массив, перед первым, а все меньшие и равные ему, после него.

**2. Определение структуры представления данных и операций над данными**

Массив — это структура данных, представленная в виде группы ячеек одного типа, объединенных под одним единым именем. Массивы используются для обработки большого количества однотипных данных. Бывают динамическим (память выделяется в процессе работы программы под указанное количество элементов) и статическими (память выделяется заранее под константное количество элементов).

Указатель – переменная, значением которой является адрес ячейки памяти. То есть указатель ссылается на блок данных из области памяти, причём на самое его начало. Указатель может ссылаться на переменную или функцию.

**3. Разработка программы**

**3.1. Декомпозиция**

Список подзадач:

1) Ввод массива

2) Вывод массива

3) Сортировка массива относительно первого элемента в соответствии с заданием

4) Проверка, является ли число степенью двойки

5) Поверка, является ли число палиндромом

6) Вставка элемента в заданную позицию со сдвигом

**3.2. Определение функций**

void output — вывод массива, получает указатель на массив и его размер.

void handmas — осуществляет ручной ввод массива с клавиатуры (используется мною только в динамическом массиве).

int PowerOfTwo — проверяет, является ли переданное в функцию число степенью двойки путем циклического деления.

void proverkaTwo — осуществляет проверку каждого элемента массива с помощью функции PowerOfTwo.

int Polindrom — проверяет, является ли переданное в функцию число палиндромом.

void proverkaPolindrom — осуществляет проверку каждого элемента массива с помощью функции Polindrom.

void sortirovka — сортирует массив по убыванию.

void insert—вставка элемента в массив со сдвигом

**3.3. Код разработанного приложения**

**3.3.1. Статический массив**

#include <iostream>

#include <cstdlib>

using namespace std;

const int N = 15;

void output(int \*mas, int n) {

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

{

cout << "Элемент массива [" << i + 1 << "]= " << mas[i] << endl;

}

}

int PowerOfTwo(int n)

{

int InitValue = 1;

while (InitValue < n)

InitValue \*= 2;

if (InitValue == n)

return 1;

else

return 0;

}

void proverkaTwo(int \*mas, int n) {

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

int k = 0;

k = PowerOfTwo(mas[i]);

if (k == 1) {

cout << mas[i] << " - степень 2. Введите новое число: ";

cin >> mas[i];

}

}

}

int Polindrom(int num1)

{

int a, b;

if (num1 > 10 && num1 < 100)

{

a = num1 / 10;

b = num1 % 10;

if (a == b)

{

return 1;

}

}

if (num1 > 100 && num1 < 256)

{

a = num1 / 100;

b = num1 % 10;

if (a == b)

{

return 1;

}

}

}

void proverkaPolindrom(int \*mas, int n) {

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

int k = 0;

k = Polindrom(mas[i]);

if (k == 1) {

cout << mas[i] << " - палиндром" << endl;

}

}

}

void sortirovka(int \*mas, int n) {

for (int i = 0; i < n - 1; i++)

{

for (int j = i + 1; j < n; j++)

{

if (mas[i] < mas[j])

{

int temp = mas[i];

mas[i] = mas[j];

mas[j] = temp;

}

}

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

int mas[N] = { 1, 4, 5, 32, 23, 55, 128, 63, 37, 42, 9, 100, 87, 222, 71 };

cout << "Исходный массив: " << endl;

output(mas, N);

cout << endl;

proverkaTwo(mas, N);

cout << "Обновленный массив: " << endl;

output(mas, N);

cout << endl;

proverkaPolindrom(mas, N);

cout << endl;

cout << "Массив, отсортированный отностительно " << mas[0] << ": " << endl;

sortirovka(mas, N);

output(mas, N);

cout << endl;

system("Pause");

return 0;

}

**3.3.2. Динамический массив**

#include <iostream>

#include <stdio.h>

#include <malloc.h>

#include <cstdlib>

using namespace std;

void output(int \*mas, int n) {

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

{

cout << "Элемент массива [" << i + 1 << "]= " << mas[i] << endl;

}

}

void handmas(int \*mas, int sizemas) {

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

{

cout << "Элемент массива [" << i << "]= ";

cin >> mas[i];

if (mas[i] > 256) cout << "Неверный размер: < 256." << endl;

while (mas[i] > 256)

{

cout << "Элемент массива [" << i << "]= ";

cin >> mas[i];

if (mas[i] > 256) cout << "Неверный размер: < 256." << endl;

}

if (mas[i] <1) cout << "Неверный размер; > 0." << endl;

while (mas[i] < 1)

{

cout << "Элемент массива [" << i << "]= ";

cin >> mas[i];

if (mas[i] > 256) cout << "Неверный размер: < 256." << endl;

}

}

}

int PowerOfTwo(int n)

{

int InitValue = 1;

while (InitValue < n)

InitValue \*= 2;

if (InitValue == n)

return 1;

else

return 0;

}

void proverkaTwo(int \*mas, int n) {

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

int k = 0;

k = PowerOfTwo(mas[i]);

if (k == 1) {

cout << mas[i] << " - степень 2. Введите новое число: ";

cin >> mas[i];

}

}

}

int Polindrom(int num1)

{

int a, b;

if (num1 > 10 && num1 < 100)

{

a = num1 / 10;

b = num1 % 10;

if (a == b)

{

return 1;

}

}

if (num1 > 100 && num1 < 256)

{

a = num1 / 100;

b = num1 % 10;

if (a == b)

{

return 1;

}

}

}

void proverkaPolindrom(int \*mas, int n) {

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

int k = 0;

k = Polindrom(mas[i]);

if (k == 1) {

cout << mas[i] << " - палиндром" << endl;

}

}

}

void sortirovka(int \*mas, int n) {

for (int i = 0; i < n - 1; i++)

{

for (int j = i + 1; j < n; j++)

{

if (mas[i] < mas[j])

{

int temp = mas[i];

mas[i] = mas[j];

mas[j] = temp;

}

}

}

}

void insert(int \*mas, int \*sizemass, int position, int elem) {

mas = (int \*)realloc((void \*)mas, (N = N + 1) \* sizeof(int));

for (int i = N; i > index - 1; i--) {

mas[i] = mas[i - 1];

}

mas[index - 1] = elem;

}

int main()

{

setlocale(LC\_ALL, "Russian");

int N, elem, index;

cout << "Введите размер массива: ";

cin >> N;

if (N < 1 || cin.fail())

cout << "Неверно. Введите число от 1." << endl;

cin.clear();

cin.ignore(cin.rdbuf()->in\_avail());

while (N<1) {

cout << "Введите размер массива: ";

cin >> N;

if (N<1 || cin.fail()) cout << "Неверно. Введите число от 1." << endl;

cin.clear();

cin.ignore(cin.rdbuf()->in\_avail());

}

int \*mas;

mas = new int[N];

handmas(mas, N);

cout << endl;

cout << "Введите элемент для замены: ";

cin >> elem;

cout << "Введите позицию: ";

cin >> index;

insert(mas, N, index, elem);

proverkaTwo(mas, N);

cout << "Обновленный массив: " << endl;

output(mas, N);

cout << endl;

proverkaPolindrom(mas, N);

cout << endl;

cout << "Массив, отсортированный отностительно " << mas[0] << ": " << endl;

sortirovka(mas, N);

output(mas, N);

cout << endl;

system("Pause");

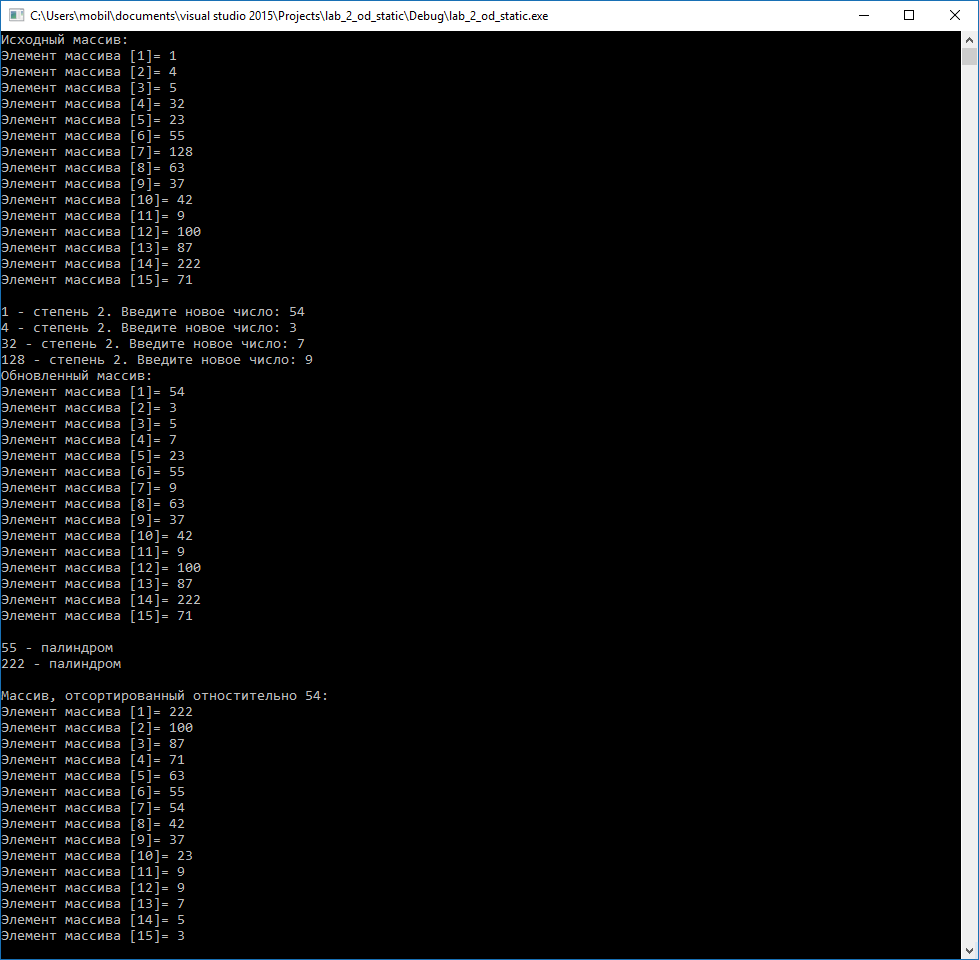
return 0;

}

**4. Тестирование программы**

**Контрольные прогоны программы**

Тест №1 — работа программы со статическим массивом.



Тест №2 — работа программы с динамическим массивом.

