Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

Пермский национальный исследовательский политехнический университет

Кафедра «Информационные технологии и автоматизированные системы»

Лабораторная работа №3

«Работа с одномерными массивами»

Вариант 23

Выполнил студент группы ИВТ-20-1Б

Фадеев Сергей Александрович

Проверил старший преподаватель кафедры ИТАС

Денис Владимирович Яруллин

Пермь 2019

**Цель:** Получение практических навыков при работе с массивами.

**Постановка задачи**

1) Сформировать массив из n элементов с помощью датчика случайных чисел (n задается пользователем с клавиатуры).

2) Распечатать полученный массив.

3) Выполнить удаление указанных элементов из массива.

4) Вывести полученный результат.

5) Выполнить добавление указанных элементов в массив.

6) Вывести полученный результат.

7) Выполнить перестановку элементов в массиве.

8) Вывести полученный результат.

9) Выполнить поиск указанных в массиве элементов и подсчитать количество сравнений, необходимых для поиска нужного элемента.

10) Вывести полученный результат.

11) Выполнить сортировку массива указанным методом.

12) Вывести полученный результат.

13) Выполнить поиск указанных элементов в отсортированном массиве и подсчитать количество сравнений, необходимых для поиска нужного элемента.

14) Вывести полученный результат.

**Вариант 23**

1. Удалить все четные элементы

2. Добавление N элементов, начиная с номера К

3. Перестановка: Сдвинуть циклически на M элементов вправо

4. Поиск: Элемент с заданным ключом (значением)

5. Сортировка: Простой выбор

**Анализ задачи**

Используем пространство имен std для работы со строками. Инициализируем массив, с заранее заданным количеством элементов.

Инициализируем функцию main и все необходимые переменные для работы программы.

Введем n, при этом n меньше размера массива и больше нуля.

Заполним массив случайными числами от -50 до 50 и выведем

Удалим четные числа из массива

Добавим в массив N элементов, начиная с номера K

Введем номер элемента, который нужно добавить. Выполним сдвиг элементов так, чтобы элемент с указанным номером оказался пустым. Присвоим ранее указанному элементу случайное значение от -49 до 50. Выведем полученный массив.

Сделаем перестановку на сдвиг, который введем.

Найдем элемент с заданным ключом(значением). Если мы нашли нужный нам элемент, то выходим из цикла и выводим успех, добавляя количество сравнений. Если такой элемент не нашелся, то выводим неудачу.

Выполним сортировку.

Инициализируем функцию для простой сортировки. Принцип простой сортировки:

Идем от i=1 до x-1, где x-количество элементов в массиве, элемента массива

Записываем в переменную min номер минимального элемента

Запускаем цикл от i+1 до x и ищем номер элемента, который будет меньше элемента с номер min

Делаем перестановку этих двух элементов

Выведем полученный массив.

Выполним поиск в сортированном массиве, тем же способом, что описан выше.

Завершим работу программы.

**Код.**

#include <iostream>

#include <ctime>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RUS");

int n, m, j, x;

srand(time(0));

cout << "Введите кол-во элементов: " << endl;

cin >> n;

int a[1000];

cout << "Рандомный массив: " << endl;;

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

{

a[i] = rand() % 100 - 50; //Рандом числа для массива

}

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

{

cout << a[i] << " "; //выводится изначальный массив

}

cout << endl << endl;

//Удаление

cout << "Удаление четных чисел из массива" << endl;

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

{

if (a[i] % 2 == 0)

a[i] = a[i + 1];

n--;

cout << a[i] << ' ';

}

cout << '\n';

//добавление

cout << "\_\_Добавление в массив N элементов, начиная с заданного номера K\_\_" << endl;

int k, N;

cout << "Введите номер элемента K: ";

cin >> k;

cout << "Введите количество элементов M:";

cin >> m;

for (int i = k; i <= n + m; i++)

{

a[i + n] = a[i];

}

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

{

a[i] = rand() % 100 - 50;

}

cout << "Массив, полученный после добавления заданных элементов: " << endl;

x = m + n;

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

{

cout << a[i] << ' ';

}

cout << endl << endl;

//Перестановка

int M, tmp;

cout << "Введите сдвиг:" << endl;

cin >> M;

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

{

for (j = x - 1; j >= 0; j--)

if (j == x - 1)

tmp = a[x - 1];

else

a[j + 1] = a[j];

a[0] = tmp;

}

cout << "Новый массив:" << endl;

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

cout << a[i] << ' ';

//Поиск 1

cout << "Поиск элемента с заданным ключом(значением) " << endl;

int kk = 0, s = 0;

cout << "Элемент: ";

cin >> k;

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

{

kk++;

if (a[i] == k)

{

s = i;

break;

}

}

if (s == 0) cout << "не найдено элементов с заданным значением k";

else

{

cout << "Элемент найден" << endl;

cout << "Количество сравнений, проведенных в ходе поиска: " << kk;

}

cout << endl << endl;

//Сортировка

cout << "Сортированный массив:" << endl;

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

{

int temp;

int min = i;

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

{

if (a[j] < a[min])

min = j;

}

temp = a[i];

a[i] = a[min];

a[min] = temp;

}

// Теперь, когда весь массив отсортирован - выводим его на экран

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

cout << a[i] << ' ';

//Поиск 2 в сортированном массиве

cout << "Поиск элемента с заданным ключом(значением) " << endl;

kk = 0, s = 0;

cout << "Элемент: ";

cin >> k;

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

{

kk++;

if (a[i] == k)

{

s = i;

break;

}

}

if (s == 0) cout << "не найдено элементов с заданным значением k";

else

{

cout << "Элемент найден" << endl;

cout << "Количество сравнений, проведенных в ходе поиска: " << kk;

}

cout << endl << endl;

}

**Результат:**

