**МИНЕСТРЕСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Балтийский государственный технический университет «ВОЕНМЕХ»

им. Д.Ф. Устинова»

Кафедра О7

«Информационные системы и программная инженерия»

Практическое задание № 5

по дисциплине «Информационные технологии и программирование»

на тему «Одномерные массивы»

Выполнила:

Студент(ка) Карпинская Полина Анатольевна

Группа Е721б

Преподаватель: Удовиченко Андрей Сергеевич,

канд. физ.-мат. наук, доцент

Санкт-Петербург

2023

**Цель работы** – познакомиться с организацией одномерных массивов, изучить принципы работы с массивами, освоить работу с массивами через указатели.

Скрипт practical\_task\_5\_1.c:

#include <stdio.h>

int main() {

int A, B;

int C[17];

printf("Enter the range values A and B: ");

scanf("%d%d", &A, &B);

printf("Enter the array elements:\n");

int count = 0;

int index = 0;

for (int i = 0; i < 17; i++) scanf("%d", &C[i]);

for (int i = 0; i < 17; i++) {

if (C[i] >= A && C[i] <= B) {

if (count == 0) {

index = i;

count++;

} else if (count == 1) {

int temp = C[i];

C[i] = C[index];

C[index] = temp;

count++;

break;

}

}

}

printf("New array: ");

for (int i = 0; i < 17; i++) printf("%d ", C[i]);

return 0;

}

Скрипт practical\_task\_5\_2.c:

#include <stdio.h>

int main() {

int A[35];

printf("Enter the array elements:\n");

for (int i = 0; i < 35; i++) {

if (scanf("%d", &A[i]) != 1 && getchar() != '\n') {

printf("Error! Incorrect values entered!");

return 0;

}

}

int min = A[0];

for (int i = 1; i < 35; i++) {

if (A[i] < min) min = A[i];

}

printf("Min: %d\n", min);

printf("Position: ");

for (int i = 0; i < 35; i++) {

if (A[i] == min) printf("%d ", i);

}

}

Ответы на контрольные вопросы:

1. Массив - это структура данных, которая представляет собой последовательность однотипных элементов.

2. Имя массива - это идентификатор, который используется для обращения к массиву.

3. Индекс элемента массива - это числовое значение, которое указывает на позицию элемента в массиве.

4. Чтобы обратиться к элементу массива, нужно использовать его имя, за которым следует индекс в квадратных скобках.

5. Адрес элемента массива можно получить, используя операцию взятия адреса (&).

6. Для описания указателя на начало массива нужно указать тип элемента массива, за которым следует символ \* и имя указателя.

7. Для обращения к элементу массива через указатель нужно использовать имя указателя, за которым следует индекс в квадратных скобках.

8. Нет, индекс элемента массива и его значение - это разные вещи.

9. Индекс первого элемента массива равен 0.

10. Массив можно проинициализировать, указав список значений элементов в фигурных скобках, разделенных запятыми.

11. Количество элементов массива можно не указывать при описании, если при инициализации массива заданы значения элементов.

12. Элементы массива могут быть любого типа данных, такого же типа, как и массив.

13. Над целым массивом можно производить операции чтения и записи элементов, сортировку, поиск, вычисление суммы и т.д.

14. При обработке массивов используются циклы для того, чтобы выполнить определенную операцию на каждом элементе массива последовательно.

15. Для того чтобы поменять местами два элемента массива, нужно сохранить значение одного элемента во временную переменную, затем присвоить значение другого элемента первому, а затем присвоить сохраненное значение второму элементу.

16. Индекс последнего элемента массива равен длине массива минус один. Например, если массив имеет длину 10, то индекс последнего элемента будет 9.

17. Да, возникнет ошибка при обращении к элементу массива, индекс которого больше индекса последнего элемента. Это может привести к выходу за пределы массива и вызвать ошибку "ArrayIndexOutOfBoundsException".

18. Да, может существовать массив из одного элемента. Для его определения нужно указать тип элемента массива и его значение в квадратных скобках, например: int arr[] = {1};.

19. Сортировка массива - это процесс упорядочивания элементов массива по определенному критерию, как возрастанию, так и убыванию.

20. Некоторые методы сортировки массива включают: сортировка пузырьком, сортировка выбором, сортировка вставками, быстрая сортировка, сортировка слиянием и т.д. В зависимости от задачи и требований можно выбирать подходящий метод сортировки.