

ФГБОУ ВО “Чувашский государственный университет им. И. Н. Ульянова”

Факультет: ИВТ

Кафедра: Вычислительной техники

Предмет: Объектно-ориентированное программирование

Лабораторная работа №3

Одномерные и многомерные массивы

Выполнил: студент группы ИВТ-41-20

Галкин Дмитрий

Проверил: доцент Обломов И.А.

Теория

Массив – представляет собой именованную область памяти, содержащую конечную последовательность однотипных элементов.

Одномерный массив: тип имя_массива [размерность] = {инициализация};

Многомерный массив: тип имя_массива [размерность_1] [размерность_2] = {инициализация};

Индивидуальное задание (Вариант 7)

Пункт А:

1. Номер максимального элемента массива
 - Функция *max(массив, размерность)* – находит максимальный элемент в массиве
2. Произведение элементов массива, расположенных между первым и вторым отрицательными элементами
 - Функция *multiplication(массив, размерность)* – находит произведение элементов в массиве между первым и вторым отрицательными элементами
3. Преобразовать массив так, чтобы в первой половине располагались элементы, стоящие в нечетных позициях, а во второй – элементы, стоящие в четных позициях
 - Функция *transfitmation(массив_1, массив_2, размерность_1, размерность_2)* – Преобразует массив в указанный вид в задании.

Пункт В: Для заданной матрицы размером 8 на 8, найти такие k, что k-я строка матрицы совпадает с k-м столбцом. Найти сумму элементов в тех строках, которые содержит хотя бы один отрицательный элемент.

```
#include <iostream>
```

```
int max(int *arr, int n) {  
    int max = -1000;  
    for (int i = 0; i < n; i++) {  
        if(arr[i] > max) {  
            max = arr[i];  
        }  
    }  
}
```

```
return max;  
}
```

```
void sum(int arr[][8], int i) {  
    long int sum = 0;  
    for (int j = 0; j < 8; ++j) {  
        sum += arr[i][j];  
    }  
    std::cout << i + 1 << " line = " << sum << std::endl;  
}
```

```
long int multiplication(int *arr, int n, int index_1, int index_2) {  
    long int multiplication = 1;  
    for (int i = index_1 + 1; i < index_2; i++) {  
        multiplication *= arr[i];  
    }  
  
    return multiplication;  
}
```

```

void transformation(int *arr_1, int *arr_2, int n) {
    int count = 0;
    for (int i = 1; i < n; i += 2) {
        arr_2[count] = arr_1[i];
        count++;
    }
    for (int i = 0; i < n; i += 2) {
        arr_2[count] = arr_1[i];
        count++;
    }
}

void show (int *arr, int n) {
    for (int i = 0; i < n; i++) {
        std::cout << arr[i] << ' ';
    }
    std::cout << std::endl;
}

int main() {
    system("chcp 1251");
    std::cout << "/****** Exercize 1 *****/" << std::endl;
    int n;
    int countNegative = 0;
    bool flag = false;
    int index_1 = 0, index_2 = 0;
    std::cout << "Enter the size of the array: "; std::cin >> n;

    std::cout << "Enter an array: ";
    int *arr_1 = new int[n];
    for (int i = 0; i < n; i++) {
        std::cin >> arr_1[i];
        if (arr_1[i] < 0) {
            countNegative++;
        }
        if(arr_1[i] < 0 && flag == true) {
            index_2 = i;
            flag = false;
        }
        if(arr_1[i] < 0 && flag == false && countNegative == 1){
            index_1 = i;
            flag = true;
        }
    }

    // Paragraph A
    std::cout << "Maximum array element: " << max(arr_1, n) << std::endl;

    // Paragraph B
    if(countNegative >= 2) {
        std::cout << "Multiplication of array elements: " << multiplication(arr_1, n, index_1, index_2) <<
std::endl;
    } else {
        std::cout << "There are no negative elements in the array" << std::endl;
    }

    // Paragraph C
    int *arr_2 = new int[n];
    transformation(arr_1, arr_2, n);
    std::cout << "Array: "; show(arr_1, n);
}

```

```

std::cout << "Transform array: "; show(arr_2, n);
std::cout << std::endl;

// Delete
delete [] arr_1;
delete [] arr_2;

std::cout << "/****** Exercize 2 *****/" << std::endl;

int arr[8][8];
countNegative = 0;
std::cout << "\t\t\tEnter an array" << std::endl;
for (int i = 0; i < 8; i++) {
    for (int j = 0; j < 8; ++j) {
        countNegative++;
        std::cin >> arr[i][j];
    }
}

// Paragraph 1
std::cout << "k*k: ";
for (int i = 0; i < 8; ++i) {
    std::cout << arr[i][i] << ' ';
}
std::cout << std::endl;

// Paragraph 2
if(countNegative == 0) {
    std::cout << "There are no negative elements in the array" << std:: endl;
} else {
    for (int i = 0; i < 8; ++i) {
        for (int j = 0; j < 8; ++j) {
            if(arr[i][j] < 0) {
                sum(arr, i);
                break;
            }
        }
    }
}
}
}

```

Вывод: Я практическим путем научился работать с одномерным и многомерным массивами.