

**Министр науки и высшего образования Российской
Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

**«Национальный исследовательский университет
ИТМО»**

**Факультет информационных технологий и
программирования**

Лабораторная работа №

Название работы

Выполнила студентка группы № М31ХХ

Фамилия Имя Отчество

Подпись:

Проверил:

Повышев Владислав Вячеславович

Санкт-Петербург

Решение с комментариями:

1. Подключен заголовочный файл <stdio.h>, отвечающий за ввод и вывод данных.
2. Весь код находится в функции main()
3. Строка scanf(...) запоминает значения, введенные в консоль. Оператор '&' это оператор взятия адреса. Например, &a означает адрес переменной 'a'.
4. Строка printf() отвечает за вывод значения.

Задание №1: Создать массив с использованием указателя.

1. int arr[4] = {0, 3, 5, 7}
Массив arr с 4 целочисленными элементами: 0, 3, 5, 7.
2. for(int i = 0; i < 4; i++) {...}
Вложенный цикл для вывода элемента массива.
3. i выводит номер элемента, начиная с нуля.
4. *(mas + i)
Указатель выводит элемент из массива <mas>, где i – индекс самого элемента.

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int mas[4] = {0, 3, 5, 7};
    int *arr = mas;
    printf("Задание 1: Массив, созданный с использованием указателя\n");
    printf("Индекс | Значение\n");
    for (int i = 0; i < 4; i++) {
        printf("\t%d | \t%d\n", i, *(arr + i));
    }
    return 0;
}
```

STDIN:

STDOUT:

Задание 1: Массив, созданный с использованием указателя

Индекс		Значение
0		0
1		3
2		5
3		7

1. `int *dynmas = (int *)malloc(4 * sizeof(int));`
Массив с 4 пустыми местами в динамической памяти.
2. `if (dynmas == NULL) {...}`
Если в массиве пусто, то выдается ошибка выделения памяти.
3. `dynmas[0] = 0; dynmas[1] = 3; dynmas[2] = 5; dynmas[3] = 7;`
Заполнение массива: 0, 3, 5, 7.
4. `for(int i = 0; i < 4; i++) {...}`
Вложенный цикл для вывода элемента массива.
5. `i` выводит номер элемента, начиная с нуля.
6. `*(dynmas + i)`
Указатель выводит элемент из массива `<dynmas>`, где `i` – индекс самого элемента.
7. `free(dynmas)`
Освобождение памяти, который был ранее выделен вызовом функции `malloc`.

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int *dynmas = (int *)malloc(4 * sizeof(int));
    if (dynmas == NULL) {
        printf("Ошибка выделения памяти.\n");
        return 1;
    }

    dynmas[0] = 0; dynmas[1] = 3; dynmas[2] = 5; dynmas[3] = 7;

    printf("\nЗадание 2: Массив в динамической памяти\n");
    printf("Индекс | Значение\n");
    for (int i = 0; i < 4; i++) {
        printf("\t%d | \t%d\n", i, *(dynmas + i));
    }
    free(dynmas);
    return 0;
}
```

STDIN:

STDOUT:

Задание 1: Массив, созданный с использованием указателя

Индекс | Значение

0		0
1		3
2		5
3		7