



Передача массива в функцию, рекурсия

практика 11

```
void printArray3 (int **a, int nrows, int ncols){
    for (int i = 0; i < nrows; i = i + 1) {
        for (int j = 0; j < ncols; j = j + 1) {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
}
```

```
int main() {
    int **a = (int**)calloc(sizeof(int*), ROWS);
    for (int i = 0; i < ROWS; i = i + 1) {
        b[i] = (int*)calloc(sizeof(int), COLS);
    }
    printArray3(a, ROWS, COLS);
}
```

Возврат массива (указателя) из функции

```
int * getArray(int n) {  
    int *array = (int*)calloc(sizeof(int), n);  
    return array;  
}  
void main (void) {  
    int arraySize = 10;  
    int *arrayFromFunc = getArray(arraySize);  
    for (int i = 0; i < arraySize; i++) {  
        printf ("%d ", arrayFromFunc[i]);  
    }  
    free(arrayFromFunc);  
}
```

Задание 1.1

Напишите две функции:

- первая возвращает максимальный элемент массива;
- вторая возвращает минимальный элемент массива.

Выведите результат работы функций на экран.

Создайте два статических массива разной длины и вызовите для каждого обе функции.

Задание 1.2

Напишите функцию, которая принимает строку и возвращает количество вхождения определенного символа в эту строку.

Какой символ искать спросите у пользователя.

Рекурсивная функция

```
void recursion(int i) {  
    printf("%d", i);  
    if (i == 0) return;  
    recursion(i - 1);  
}
```

```
void main (void) {  
    recursion(10);  
}
```

Задание 2.1

Напишите рекурсивную функцию, которая выводит на экран числа от 'a' до 'b'.

Числа 'a' и 'b' спросите у пользователя.



Задание 2.2

Напишите рекурсивную функцию, которая находит сумму ЦИФР числа. Число спросите у пользователя.

Примерная схема метода:

```
int функция(параметр) {  
    if (базовый_случай) return при_базовом_случае;  
    return (сохраняемая_часть) + функция(след._случай);  
}
```