

**Министерство науки и образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
"Московский институт электронной техники"  
(МИЭТ)**

**Отчет по лабораторной работе № 2А**

**" Ввод-вывод при помощи libc"**

**Выполнили:** студенты ПМ - 31

Алтухова Анна Валерьевна

Мартынова Мария Олеговна

## Задание 2А.1

### Задание:

Разработайте программу на языке C/C++, создающую, инициализирующую одинаковыми значениями и выводящую на экран при помощи функции `libc printf()` массивы из  $N = 5$  чисел:

- *Ms* из 16-битных целых чисел (0xC0DE);
- *Ml* из 32-битных целых чисел (0x DEAD BEEF);
- *Mq* из 64-битных целых чисел (0x BECA CA01 FFED C0CA);
- *Mfs* из 32-битных чисел с плавающей запятой (значение *x* по варианту);
- *Mfl* из 64-битных чисел с плавающей запятой (значение *x* по варианту).

Выведите каждый из целочисленных массивов как в знаковом десятичном (*d*), так и в шестнадцатеричном (*X*) виде, чтобы убедиться, что короткие значения не расширены до 32 бит, а длинные — не усечены.

### Листинг:

```
#include <iostream>

int main() {
    const int16_t i16 = (int16_t)0xC0DE;
    int16_t Ms[5] = {i16, i16, i16, i16, i16};
    printf("%#hx, %#hx, %#hx, %#hx, %#hx\n", Ms[0], Ms[1], Ms[2], Ms[3],
Ms[4]);
    printf("%hi, %hi, %hi, %hi, %hi\n", Ms[0], Ms[1], Ms[2], Ms[3], Ms[4]);
    const int32_t i32 = (int32_t)0xDEADBEEF;
    int32_t Ml[5] = {i32, i32, i32, i32, i32};
    printf("%#x, %#x, %#x, %#x, %#x\n", Ml[0], Ml[1], Ml[2], Ml[3], Ml[4]);
    printf("%d, %d, %d, %d, %d\n", Ml[0], Ml[1], Ml[2], Ml[3], Ml[4]);
    const int64_t i64 = (int64_t)0xBECACA01FFEDC0CA;
    int64_t Mq[5] = {i64, i64, i64, i64, i64};
    printf("%#lx, %#lx, %#lx, %#lx, %#lx\n", Mq[0], Mq[1], Mq[2], Mq[3],
Mq[4]);
    printf("%ld, %ld, %ld, %ld, %ld\n", Mq[0], Mq[1], Mq[2], Mq[3], Mq[4]);
    const float f32 = 3.14159265358979323846f;
    float Mfs[5] = {f32, f32, f32, f32, f32};
    const float f64 = 3.14159265358979323846;
    float Mfl[5] = {f64, f64, f64, f64, f64};
}
```

### Пример вывода:

```
0xc0de, 0xc0de, 0xc0de, 0xc0de, 0xc0de
-16162, -16162, -16162, -16162, -16162
0xdeadbeef, 0xdeadbeef, 0xdeadbeef, 0xdeadbeef, 0xdeadbeef
-559038737, -559038737, -559038737, -559038737, -559038737
0xbecaca01ffedc0ca, 0xbecaca01ffedc0ca, 0xbecaca01ffedc0ca,
0xbecaca01ffedc0ca, 0xbecaca01ffedc0ca
-4698721151270141750, -4698721151270141750, -4698721151270141750, -
4698721151270141750, -4698721151270141750
```

## Задание 2А.2:

### Задание:

Для каждого массива  $M$  введите с клавиатуры новое значение элемента  $M[i]$ ,  $i = 2$  при помощи функции libc `scanf()`. Проанализировав возвращённое `scanf()` значение, определите корректность ввода; при необходимости отобразите сообщение об ошибке при помощи функции libc `puts()`. Выведите массивы на экран снова, убедитесь, что элемент  $M[i]$  приобрёл ожидаемое значение, а другие элементы массива не изменились.

Примечание: В данном задании необходимо передать функции `scanf()` адрес  $M[i]$ , а не промежуточной переменной — иначе нет смысла контролировать значение соседних элементов

### Листинг:

```
#include <iostream>

int main() {
    const float f32 = 3.14159265358979323846f;
    float Mfs[5] = {f32, f32, f32, f32, f32};
    const double f64 = 3.14159265358979323846;
    double Mfl[5] = {f64, f64, f64, f64, f64};
    if(scanf("%f", &Mfs[2]) != 1)
    {
        puts("Bad float\n");
    }
    printf("%f, %f, %f, %f, %f\n", Mfs[0], Mfs[1], Mfs[2], Mfs[3], Mfs[4]);
    if(scanf("%lf", &Mfl[2]) != 1)
    {
        puts("Bad double\n");
    }
    printf("%lf, %lf, %lf, %lf, %lf\n", Mfl[0], Mfl[1], Mfl[2], Mfl[3],
Mfl[4]);
}
```

### Пример вывода:

```
123
3.141593, 3.141593, 123.000000, 3.141593, 3.141593
```

342

3.141593, 3.141593, 342.000000, 3.141593, 3.141593

### Задание 2А.3

#### Задание:

Для одного из массивов (по варианту)  $M$  введите с клавиатуры новое значение всех пяти элементов при помощи одного вызова функции `libc scanf()`. Проанализировав возвращённое `scanf()` значение, определите корректность ввода; при необходимости отобразите сообщение о количестве не заданных элементов.

Выведите массив на экран снова.

#### Листинг:

```
#include <iostream>

int main() {
    const int16_t i16 = (int16_t)0xC0DE;
    int16_t Ms[5] = {i16, i16, i16, i16, i16};
    auto ir = scanf("%hi %hi %hi %hi %hi", &Ms[0], &Ms[1], &Ms[2], &Ms[3],
&Ms[4]);
    if(ir != 5)
    {
        printf("Error in %d elements", 5 - ir);
    }
}
```

#### Пример вывода:

3 e 4 5 6

Error in 4 elements

### Задание 2А.4

#### Задание:

Введите с клавиатуры при помощи функций `libc`:

- слово (строку без пробелов)  $s1$ ;
- строку, возможно, содержащую пробелы  $s2$ ;
- слово  $s3$  таким образом, чтобы принимающий его буфер гарантированно не переполнился (если буфер длины  $k$  — вводить не более  $k - 1$  символов, при необходимости добавляя завершающий нулевой символ).

Выведите на экран при помощи функций `libc` строки «\*\*\*  $s1$  \*\*\*», «\*\*\*  $s2$  \*\*\*», «\*\*\*  $s3$  \*\*\*» (между звёздочками должна быть введённые строки, а не литералы  $s1$ - $s3$ ).

### ***Листинг:***

```
#include <stdio.h>

int main()
{
    char s1[16];
    gets(s1);
    char s2[32];
    gets(s2);
    char s3[32];
    fgets(s3, 32, stdin);
    printf("«*** %s ***», «*** %s ***», «*** %s ***»", s1, s2, s3);
}
```

### ***Пример вывода:***

```
ddsavasf
dgjn dfdfg asd
ergndsfnkjdnkndskvjbskfdjbnjksdnvkjnzskvjzxnckjvnzlkknkzjdsvnkijnalksdnvkjdsnv
ldsnlv
«*** ddsavasf ***», «*** dgjn dfdfg asd ***», «***
ergndsfnkjdnkndskvjbskfdjbnjksd ***»
```

### ***Задание 2А.5***

#### ***Задание:***

Выведите на экран при помощи функций `libc` массивы  $M_s \dots M_{fl}$  как таблицу из пяти строк и  $N$  столбцов (младшая цифра под младшей цифрой).

### ***Листинг:***

```
#include <iostream>

int main()
{
    const int16_t i16 = (int16_t)0xC0DE;
    int16_t Ms[5] = {i16, i16, i16, i16, i16};
    const int32_t i32 = (int32_t)0xDEADBEEF;
    int32_t Ml[5] = {i32, i32, i32, i32, i32};
    const int64_t i64 = (int64_t)0xBECACA01FFEDC0CA;
    int64_t Mq[5] = {i64, i64, i64, i64, i64};
    const float f32 = 3.14159265358979323846f;
    float Mfs[5] = {f32, f32, f32, f32, f32};
    const float f64 = 3.14159265358979323846;
    float Mfl[5] = {f64, f64, f64, f64, f64};
    printf("");
}
```

```

    printf("%20hi %20hi %20hi %20hi %20hi\n", Ms[0], Ms[1], Ms[2], Ms[3],
Ms[4]);
    printf("%20i %20i %20i %20i %20i\n", Ml[0], Ml[1], Ml[2], Ml[3], Ml[4]);
    printf("%20li %20li %20li %20li %20li\n", Mq[0], Mq[1], Mq[2], Mq[3],
Mq[4]);
    printf("%20f %20f %20f %20f %20f\n", Mfs[0], Mfs[1], Mfs[2], Mfs[3],
Mfs[4]);
    printf("%20lf %20lf %20lf %20lf %20lf\n", Mfl[0], Mfl[1], Mfl[2], Mfl[3],
Mfl[4]);
}

```

### ***Пример вывода:***

-16162	-16162	-16162	-16162	-16162
-559038737	-559038737	-559038737	-559038737	-559038737
-4698721151270141750	-4698721151270141750	-4698721151270141750	-4698721151270141750	-4698721151270141750
3.141593	3.141593	3.141593	3.141593	3.141593
3.141593	3.141593	3.141593	3.141593	3.141593