

# Задание 1

Разработайте консольное приложение, обеспечивающее ввод с консоли и вывод на консоль целого числа и символа.

ВВОД	ВЫВОД
F 45	F 45

## Программа

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int a;      //объявление целого числа
    char st[10]; //объявление "строки"
    scanf("%d\n%s",&a, st); //ввод
    printf("%d\n%s",a,st);   //вывод
    return 0; }
```

# Задание 2

Разработайте программу, рассчитывающую значения двух указанных функций. Входные значения аргументов запросите с консоли. Полученные значения функций выведите на консоль.

ВВОД	ВЫВОД
22.5	1.017861 1.017861

```
#include <stdio.h>

#define _USE_MATH_DEFINES

#include <math.h>

int main()
```

```

{
    double x,z1,z2; //объявление
    scanf("%lf",&x); //ввод значений x
    z1= (cos(x) + sin(x))/(cos(x) - sin(x)); // расчёт примера 1
    z2= tan(2*x)+1/cos(x*2); //расчёт 2го.. secX=1/cosX
    printf("%lf\n%lf", z1, z2); //вывод переменных z1 z2
    //( %lf вместо %f из-за типа переменных double)
    return 0;
}

```

## Задание 3

- 1) Запросите с консоли целое число в указанной системе счисления.
- 2) Выведите на консоль введённое число в указанной системе счисления.
- 3) Выведите на консоль введённое в задании 1 число в 16-ричной или 8-ричной системе счисления, а также это же число в той же системе счисления, но сдвинутое влево/вправо на указанное число бит.
- 4) Выведите на консоль введённое в задании 1 число в 16-ричной или 8-ричной системе счисления (согласно заданию 3), а также это же число в той же системе счисления после применения к нему битовой операции отрицания.
- 5) Введите с консоли целое число в системе счисления, указанной в задании 3 Выведите на консоль результат указанной битовой операции (и, или, исключающее или) введённого числа и числа, введённого в задании 1 в системе счисления, указанной в задании 3

ВВОД	ВЫВОД
9	9 11 9 2 9 ffffffff6
9	9

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

```
int main(void)
```

```
{
```

```
    int dig,dig1; // задать 2 целых числа
```

```
    scanf("%d",&dig); //ввести первое число
```

```
    printf("%d\n",dig); //вывести на экран число в 10ой СС
```

```
    printf("%o\n",dig); //RE: в 8ой СС
```

```
    printf("%x %x\n",dig, dig>> 2); //RE: в 16ой СС,
```

```
    //+битовое значение со сдвигом 2 вправо
```

```
    printf("%x %x\n",dig, ~dig); //вывод в 16ой СС
```

```
    //+то же самое число под логическим отрицанием
```

```
    scanf("%x\n",&dig1); // ввести второе число
```

```
    printf("%x", dig & dig1); //побитовая конъюнкция
```

```
+вывод в 16ой СС
```

```
    return 0;
```

```
}
```

## Задание 4

1) Запросить с консоли целое число и проверить его на попадание в заданный диапазон значений. Результат такой проверки вывести на консоль. При выполнении задания не использовать условный оператор.

2) Запросить с консоли целое число и проверить значение бита с указанным номером в этом числе. Результат такой проверки вывести на консоль. Для удобства разрешается запрашивать указанное число в восьмеричной или шестнадцатеричной системе счисления. При выполнении задания важно обратить внимание на грамотный выбор типа данных анализируемого числа.

Ввод	Вывод
34	1
8	1

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

```
int main() {
```

```
    int dig, dig1, x; // объявление
```

```
    scanf("%d", &dig); //ввод целочисленной переменной dig  
    для №1
```

```
    x = ((dig >= 33) && (dig <= 55)); //проверка: входит ли в  
    диапазон x[33-55]
```

```
    printf("%d\n", x); //1[входит] 2[нет]
```

```
    scanf("%d", &dig1); //ввод второй переменной
```

```
    printf("%d", (dig1 >> 13) & 1); //сдвигом на 13 битов вправо  
    выбрать 14ый по порядку бит и определить его чётность.
```

```
    return 0;
```

```
}
```

*Лабораторные работы 1-4  
выполнил студент 1 курса ИС  
Микаилов Михаил Аскерович  
3110/14 вариант*