

## Лабораторная работа №1

### Числа с фиксированной точкой

#### Инструментарий и требования к работе

Допустимые языки	C	C++	Python	Java
Стандарты / версии	C17	C++20	3.12.5	Temurin-21.0.4
Требования для всех работ	<a href="#">Правила оформления и написания работ</a>			

#### Задание

Необходимо написать программу, которая позволяет выполнять арифметические действия с дробными числами в форматах с фиксированной точкой. Программа должна использовать только целочисленные вычисления и типы данных.

Аргументы программе передаются через командную строку в одном из двух вариантов:

1. `<A.B> <округление> <число>`
2. `<A.B> <округление> <число1> <операция> <число2>`

где:

- A.B – задаёт формат представления входных чисел. A и B – неотрицательные целые числа, обозначающие целую и дробную часть. Гарантируется, что  $A+B \leq 32$  и  $A \geq 1$ . Числа со знаком в дополнении до 2.
- операция – символ арифметической операции: `+`, `-`, `*`, `/`.
- число – одно или два числа, записанные в 16-ричной побитовой форме с префиксом `'0x'` в любом регистре, уместяющиеся в 32 бита.

- округление – задаёт тип округления:

тип округления	пояснение	общепринятое обозначение
0	к нулю	toward_zero
1	к ближайшему чётному	toward_nearest_even
2	к $+\infty$	toward_pos_infinity
3	к $-\infty$	toward_neg_infinity

В случае присутствия операции, её результат должен быть получен в том же формате, что и входные числа. Если в результате операции необходимо сделать округление и применить модулярную арифметику, то первым выполняется округление.

Результат операции или единственное входное число необходимо вывести в стандартный поток вывода в виде:

- десятичная запись с 3 десятичными цифрами после точки. Пример:  
**0.120**
- при делении на 0 – вывести **div\_by\_0** и завершиться с 0 кодом возврата.

Примеры:

Входные аргументы	Результат
16.12 0 0x17360	23.210
8.8 1 0xdc9f + 0xd736	-76.168
8.8 2 0xdc9F + 0xd736	-76.167
4.4 3 0x12 / 0x100	div_by_0

Если вы пишете на C++ и у вас не проходит часть тестов (рандомно), то читаем: [Does exit\(\) flush and close `ofstream` objects? - Stack Overflow](#)

### Запуск автотестов

<https://skkv-itmo.gitbook.io/ct-comp-arch-course/course-format/autotests-github>

Число попыток = 30.

*из Web-интерфейса*

4 варианта округления. Нужно выбрать подходящий (по умолчанию выбран округление к 0).

*через CLI интерфейс:*

Запуск скрипта

```
gh workflow run classroom.yml --ref main -f <field>=<value>
```

Пример

```
gh workflow run classroom.yml --ref main -f rounding="0"
```

Ключ: rounding ("Округление") default: "0"

options: "0" "1" "2" "3"