

## Lab 4.2.1 Comparing floating-point numbers

### Objectives

Familiarize the student with:

- the quirks of comparing floating-point values;
- ways to mitigate the problem of comparing said values.

### Scenario

Comparing floating-point numbers in C++ may lead to unexpected results.

Run the following program and modify the **is\_close** function to get the desired results.

You may want to use the **fabs** function from the `math.h` library.

```
#include <iostream>
#include <math.h>

using namespace std;

bool is_close(double a, double b, double tolerance){
    // Your code goes here
    return false;
}

int main(void) {

    if (0.3 == 3 * 0.1) {
        cout << "The numbers are equal";
    } else {
        cout << "The numbers are not equal";
    }
    cout << endl;

    if (is_close(0.3, 3 * 0.1, 0.00000001)) {
        cout << "The numbers are close enough";
    } else {
        cout << "The numbers are not close enough";
    }
    cout << endl;

    // this should work regardless of the argument order
    if (is_close(3 * 0.1, 0.3, 0.00000001)) {
        cout << "The numbers are still close enough";
    } else {
        cout << "The numbers are not close enough";
    }
    cout << endl;

    if (is_close(3 * 0.1, 0.31, 0.00000001)) {
        cout << "The numbers are still close enough";
    } else {
        cout << "The numbers are not close enough";
    }
    cout << endl;

    return 0;
}
```

## Example output

```
The numbers are not equal
The numbers are close enough
The numbers are still close enough
The numbers are not close enough
```