


# Code Intent, or: can I clearly express myself in C++?

(C) Richèl Bilderbeek 

March 23, 2015

# Overview

- Introduction
- Example
- Exercises

# Programming Ideals<sup>1</sup>

- Correctness
- Reliability
- Affordable
- Maintainable

---

<sup>1</sup>Stroustrup 2009, §1.6

# Introduction

Programs must be written for people to read, and only incidentally for machines to execute<sup>2</sup>

It's hard to overstate the value of simple design and clear code<sup>3</sup>

- Code has an intention
- Code should match the intention
- Your intention can be concluded from:
  - what you put in
  - what you omit

---

<sup>2</sup>H. Abelson & G.J. Sussman

<sup>3</sup>Sutter & Alexandrescu 2005, §6

## Example 1

What can be concluded from the following code?

```
int x;
```

## Example 1

What can be concluded from the following code?

```
int x;
```

### Conclusions

- x is either really an x or a short-living simple-use variable or coder is unaware of the literature
- x can both be positive or negative
- there is no value for x to initialize it with yet and coder is unaware of the literature
- x will have its value changed at least once

## Example 2

What can be concluded from the following code?

```
unsigned int x;
```

## Example 2

What can be concluded from the following code?

```
unsigned int n_countries = 27;
```

### Conclusions

- `n_countries` is probably a number of countries
- `n_countries` is always positive
- coder is unaware of the literature or some complex code is coming
- `n_countries` will have its value changed at least once