

The Little CPlPler

idea: at some point in the code explain why is the book written in English and no any other language

Prerequisites

What to expect from this book

A phrase telling them what to do with the book

Tools

Tools necessary to follow the book (for everything to work), note: the source code can be found in a repo, thought

Using the tools

Ask to open a file with a text editor

Extend explanation on how to compile

ask to modify the file and see what happens

Given the file 000-hello-world.cc:

```
#include <iostream>

int main()
{
    std::cout << "Hello World!" << std::endl;
    return 0;
}
```

compile the file into a executable with either clang++ or g++:

add std option to make it C++11 compilant

```
clang++ 000-hello-world.cc -o 000-hello-world.exe
g++ 000-hello-world.cc -o 000-hello-world.exe
```

and run it with:

```
./000-hello-world.exe
```

the output of the program in terminal will be:

Hello World!

Content

How to read this book

Yeah, you should learn first how to read this book! :P

Explain how the book is supposed to be read (try to answer question with answer covered, then reveal answer and try understand what is it doing)

Basics

Getting to know C++(11)

Add interludes asking whoever is reading to pause for a while to recover from so much info

000.

What do you think the following code will output after compiling and running it?

```
#include <iostream>

int main()
{
    std::cout << "Hello World!" << std::endl;
    return 0;
}
```

The output is:

Hello World!

001.

Now, what if you compile this other file:

```
#include <iostream>

int main()
{
    std::cout << "Hello World!" << std::endl;
    std::cout << "I'm a program example and I'm "
              << "in English."
              << std::endl;
    return 0;
}
```

The output is:

Hello World!
I'm a program example and I'm in English.

Pay close attention to the output, there are three sentences surrounded by quotation marks ("), but there are only two lines in the output. Why?

002.

If we change our example slightly (notice the semicolon (;)) what do you think it will happen?

```
#include <iostream>

int main()
{
    std::cout << "Hello World!" << std::endl;
    std::cout << "I'm a program example and I'm ";
                << "in English."
                << std::endl;
    return 0;
}
```

Well, it doesn't compile! We get an error similar to:

```
test.cc:7:13: error: expected expression
              << "in English."
              ^
```

It is telling us that it was expecting something (a `std::cout` for example) before `<<`.

Try removing or adding random characters (anywhere) to the example and you will find that the compiler just admits a certain arrangement of characters and not much more. But, why? Well, the compiler just understands the grammar of C++ as we just understand the grammar of our human languages. Going a little further with the analogy, we can understand the grammar of any human language (its parts (verbs, prepositions, ...) and how are they connected) but we can only understand the meaning (semantics) of those languages we have studied (or our mother tongues).

003.

Does the following program compile. If yes, what is its output?

```
#include <iostream>

int main()
{
    std::cout << "Hello World!" << std::endl;
    std::cout << "I'm a program"
                << std::endl
                << "example and I'm"
                << std::endl
                << "in English."
                << std::endl;
    return 0;
}
```

Yep, it in fact compiles, and its output is:

```
Hello World!
I'm a program
example and I'm
in English.
```

Notice how `std::endl` puts text in a new line, that's its whole job in fact.