

# 经典中的博弈: 第一章 C++的 Hello, World!

摘要: 原创出处: <http://www.cnblogs.com/Alandre/> 泥沙砖瓦浆木匠 希望转载, 保留摘要, 谢谢!

“程序设计要通过编写程序的实践来学习” —Brian Kernighan

## 1.1 程序

何为程序?简单的说,就是为了使计算机能够做事,你需要在繁琐的细节中告诉它怎么做。对于怎么做的描述就是程序。编程是书写和测试怎么做的过程。维基百科上说,一个程序就像一个用汉语(程序设计语言)写下的红烧肉菜谱(程序),用于指导懂汉语和烹饪手法的人(体系结构)来做这个菜。做一个形象的比喻,在生活中某种意义上,你都见到程序。石器时代,“我们用石头打它”。就是个程序。

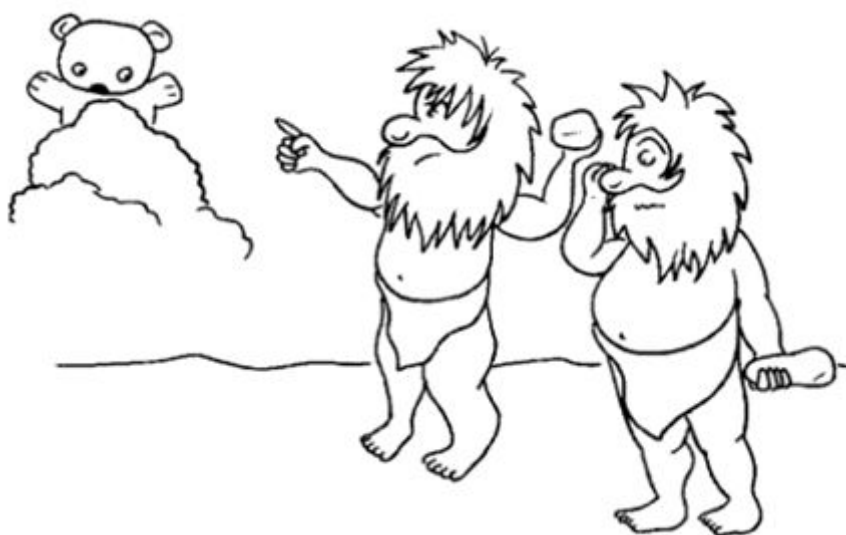


图 1.1 “我们用石头打它” 摘自 《数学之美》

其中,他用怪叫和动作告诉小伙伴这行为,叫做程序。只有懂这个的伙伴才会一起拿起石头砸。但相对计算机来说,计算机很“笨”,为了像计算机解释这些,我们需要一种特定语法精确定义的语言。那就是编程语言,例如 C++。

## 1.2 经典品味 ” Hello, world! “

这是经典的例子,在屏幕中输出”Hello, world! “:

```
1#include <iostream>
2using namespace std;
3
4int main()
5{
6    cout << "Hello,World!\n";
7    return 0;
8}
```

运行的结果:打印 Hello,World!然后,下面新的一行.一个光标位于下一行的开始。

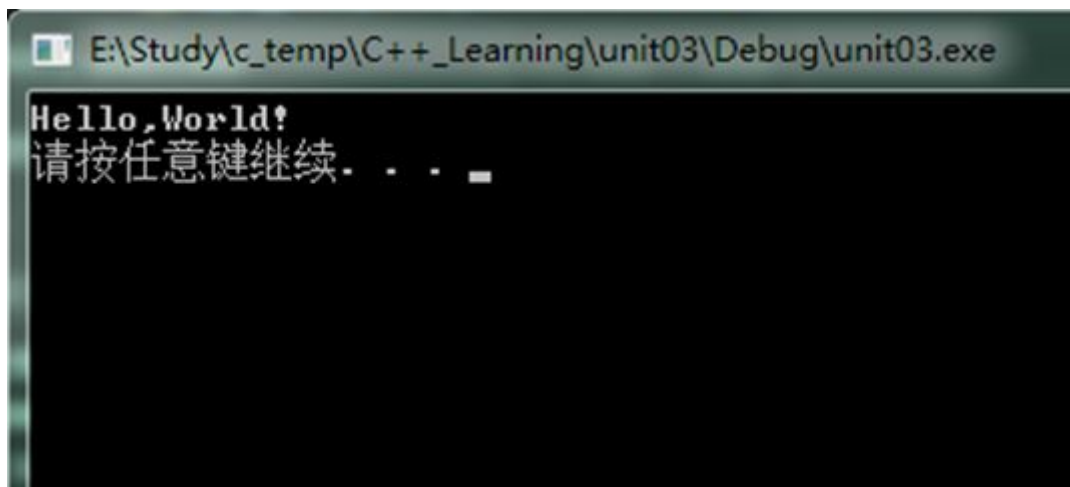


图 1.2 Hello,World!输出

程序中，主要工作的代码：

```
1cout << "Hello,World!\n";
```

`cout` 是一个标准输出流，使用 `<<` 操作符可以将字符显示到屏幕上。很自然，这是你第一次看到缩写，`cout`: `character outputstream` 的缩写。刚开始会不适应，但是当你遇见缩写多了的话，你会爱上他。

第一行：

```
1#include <iostream>
```

他是个指令，这个具体是让我们使用标准的 C++流 I/O 功能。这里，我们使用了 `cout` 和它的操作符 `<<`。此外，使用 `#include` 包含的文件通常后缀有 `.h`，称为头文件或头。我们基本上是使用 C++自己提供的各种功能源代码。比如上面，我们需要什么功能就引入啥头文件。

程序入口:

```
1 int main()  
2 {  
3     return 0;  
4 }
```

大家都知道一个程序要告诉计算机哪里开始执行，C++每个程序必须有个 `main` 函数。一个函数基本是个指令集也称指令序列，计算机会按着原定的顺序执行。一个函数通常包括:

- 返回类型: 这里是 `int`，用来指定返回结果类型。
- 名字: 这里是 `main`
- 参数列表: 这里为空，如果有参数，会写在 `()`
- 函数体: `{}` 里面写了函数要执行的语句。

程序的结束:

```
return 0;
```

要知道，`main` 函数是系统调用的，自然我们需要告诉系统何时结束。返回值可以用于检查程序是否成功，特别在有些系统 (`unix/linux`)。有 `main ()` 返回的一个 `0` 表示程序成功终止。

## 1.3 你看不到的程序代码之外：编译 链接

”Hello, world!”程序的运行，通过 C++ 编程语言。在它从人可读可改的格式转换为计算机可以理解的东西，这个过程我们需要一个成为“编译器”的程序。

何为编译器？

维基百科上说，编译器 (Compiler)，是一种电脑程序，它会将用某种编程语言写成的源代码 (原始语言)，转换成另一种编程语言 (目标语言)。它主要的目的是将便于人编写，阅读，维护的高级计算机语言所写作的源代码程序，翻译为计算机能解读、运行的低阶机器语言的程序，也就是可执行文件。做个比喻，一个声控电灯，我们如果想说“把灯打开”，灯显然是不知道这句话的意思。它的

世界是：“有声音我就亮”。进而，你的仆人就大叫一声，灯亮了。这个仆人的工作就是编译器的程序原理。

一个现代编译器的主要工作流程如下：源代码（source code）→ 预处理器（preprocessor）→ 编译器（compiler）→ 汇编程序（assembler）→ 目标代码（object code）→ 链接器（Linker）→ 可执行文件（executables）。因此，可读可写的称为源代码，计算机可执行的是目标代码。在未来的日子里，你将接触到这个朋友，他非常严格。但请记住，编译器是你编程中最好的朋友。下面我们展现程序的流程：

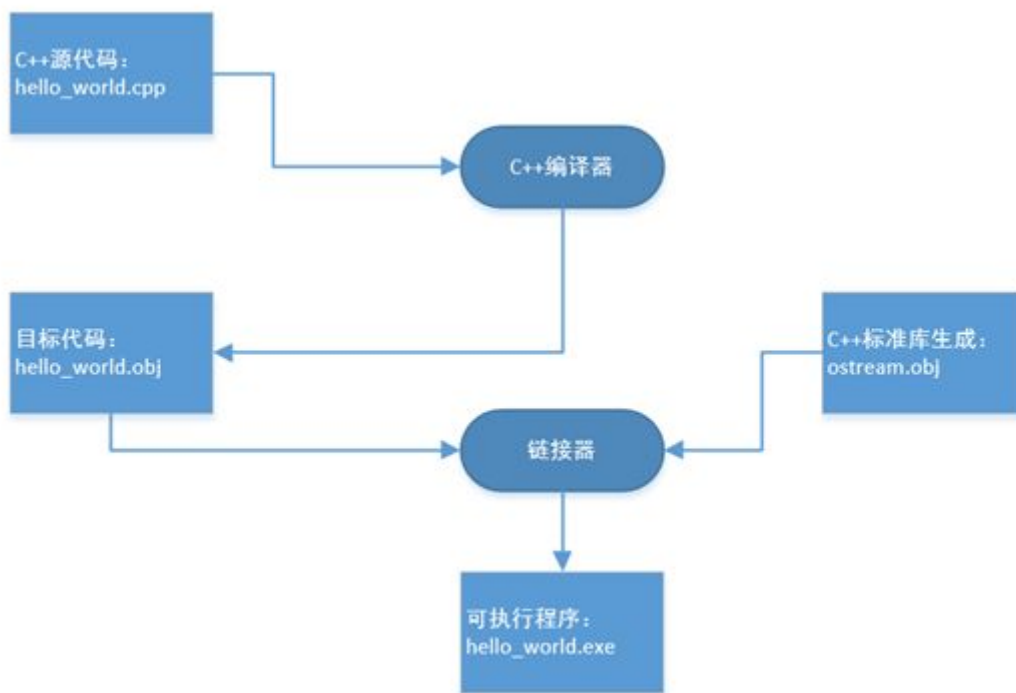


图 1.3 Hello, World! 的一生

## 1.4 第一章 小结 及 思考

Hello,World! 其实不重要，重要我们熟悉基本思想和工具。曾经看透菜谱，成为了大厨。曾经看破历史的，当了作家或者其他。勤奋，专注，锲而不舍。C++ 并不难。

如以上文章或链接对你有帮助的话，别忘了在文章按钮或到页面右下角点击“赞一个”按钮哦。你也可以点击页面右边“分享”悬浮按钮哦，让更多的人阅读这篇文章

