

C语言起源

1972年，贝尔实验室的丹尼斯·里奇（Dennis Ritchie）和肯·汤普逊（Ken Thompson）在开发UNIX操作系统时设计了C语言。然而，C语言和并不是突发奇想而来，而是在B语言的基础上进行设计。

C语言设计的初衷是将其作为程序员使用的一种编程工具，主要目标是成为有用的语言。



C语言特性

- **高效性：**相较于汇编语言C语言可读性更强，效率也接近汇编语言，可以高效利用计算机资源。
- **面向过程编程：**让程序员可以自顶向下的规划、结构化编程和模块化设计，使得解决问题更加方便。
- **可移植性较好：**C语言可以运行在不同操作系统中，然而，需要稍作修改，一般只修改一些头文件。



Powerful control structures



Fast

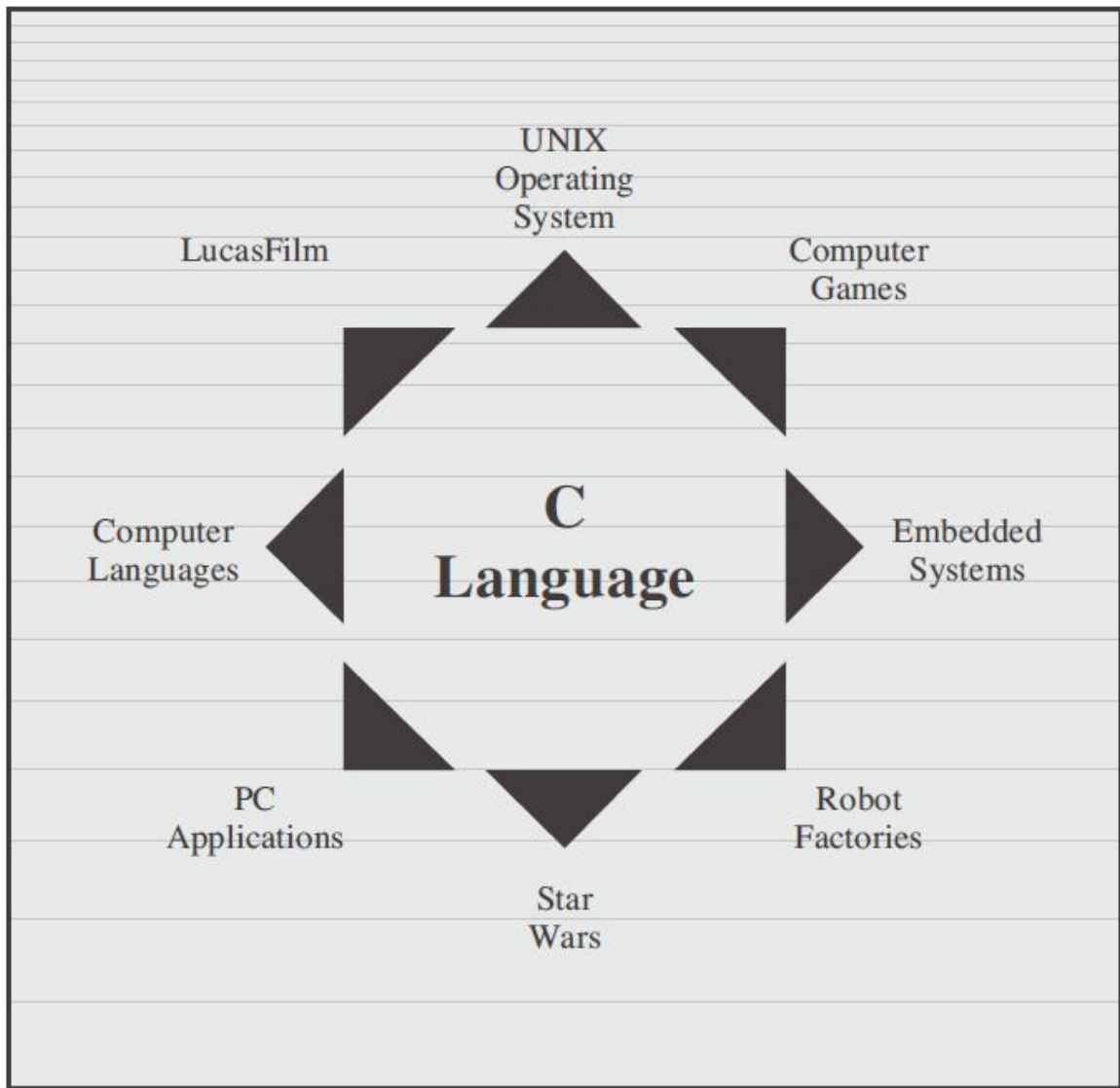


Compact code—small programs



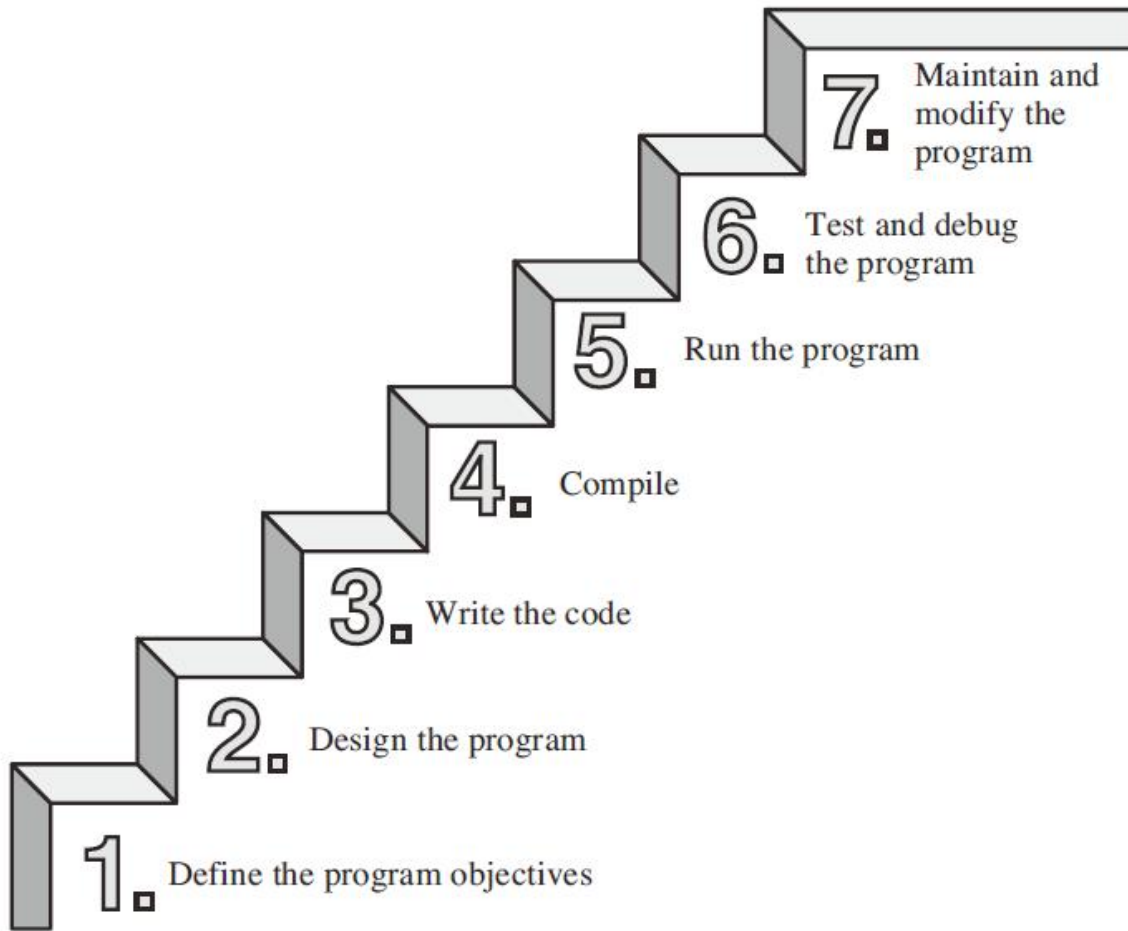
Portable to other computers

C语言应用范围：在需要和机器打交道或者需要高性能的领域，C语言是首要选择，比如操作系统、嵌入式、游戏服务器等等。

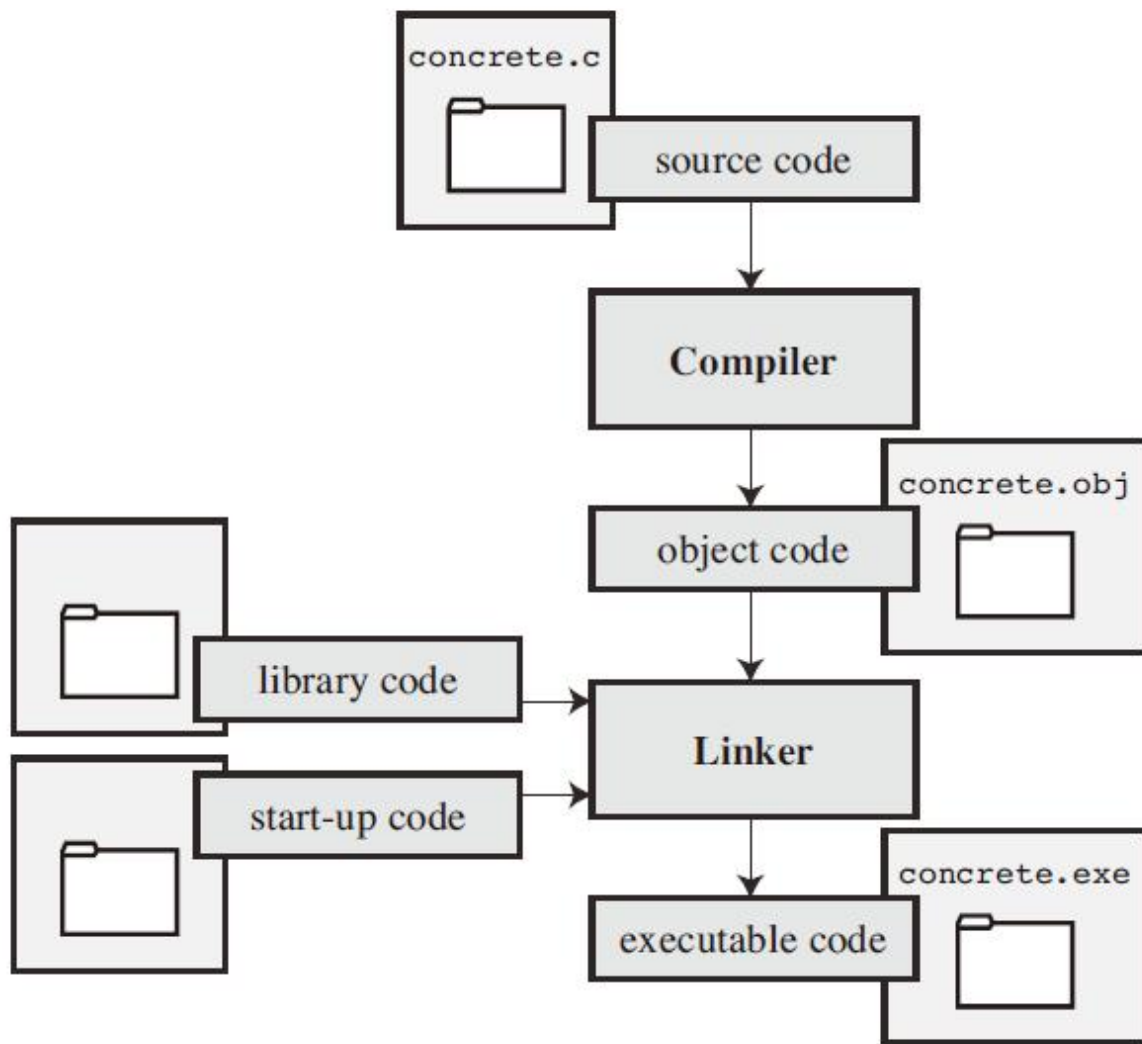


C语言和C++比起来，更加灵活，然而也更容易出错。然而，C++ 的面向对象特性更加适合大型程序设计。

使用C语言7个步骤



C编程的基本策略是，将源代码转换成可执行文件，通过编译和链接两个步骤来完成这一过程。编译器把源代码转换成中间代码，链接器把中间代码和其他代码合并，生成可执行文件。



伟大的开始——Hello,World!

伟大的开始——Hello,World!

描述

编写一个能够输出“ Hello,World!”的程序，这个程序常常作为一个初学者接触一门新的编程语言所写的第一个程序，也经常用来测试开发、编译环境是否能够正常工作。

提示：“ Hello,World!”中间没空格。

输入

无输入

输出

Hello,World!

输入样例 1

无输入

输出样例 1

Hello,World!

```
1 #include <stdio.h>
2
3 int main(void){
4     printf("Hello,world!\n");
5     return 0;
6 }
```

常用编程软件:

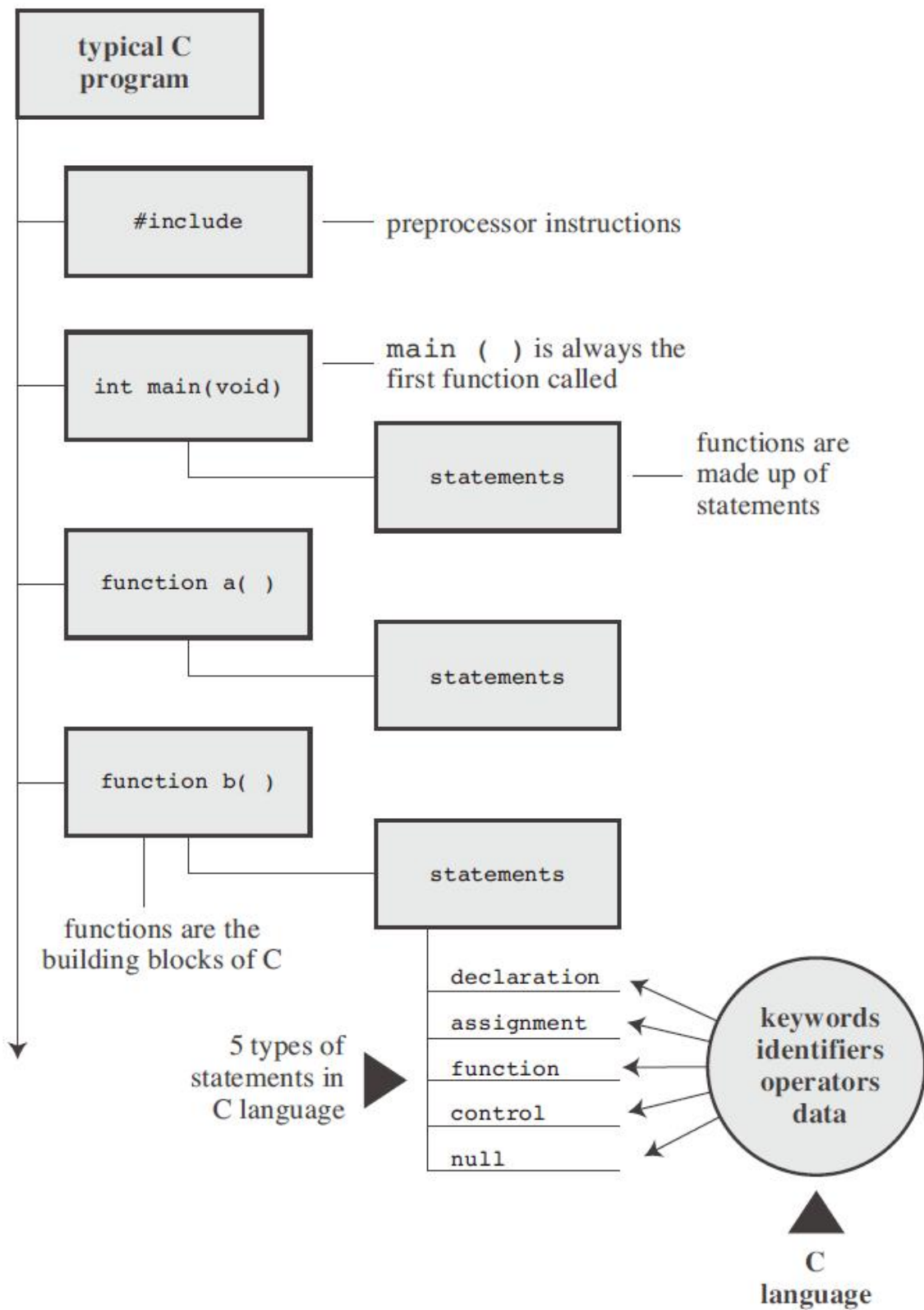
1. Dev C++: 轻量级, 使用简单
2. VS 2022: 重量级IDE, 功能强大
3. Linux 命令行: 需要掌握Linux基础命令
4. 黑猫OJ 在线IDE: 在线使用, 无需安装

一个简单的C样例

```
#include <stdio.h>
int main(void)                /* a simple program          */
{
    int num;                   /* define a variable called num */
    num = 1;                   /* assign a value to num        */

    printf("I am a simple "); /* use the printf() function    */
    printf("computer.\n");
    printf("My favorite number is %d because it is first.\n",num);

    return 0;
}
```



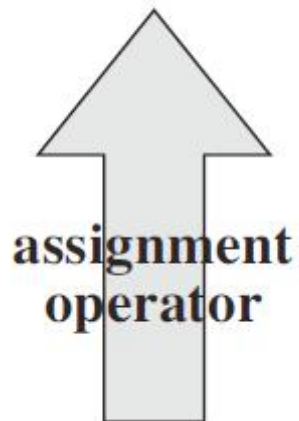
- #include是预处理指令，在编译前工作，相当于将其后的文件复制粘贴到当前文件。
- main是程序的入口。
- 注释：// 单行注释 和 /**/ 多行注释。

变量命名：

Valid Names	Invalid Names
wiggles	\$Z]**
cat2	2cat
Hot_Tub	Hot-Tub
taxRate	tax rate
_kcab	don't

赋值操作符:

```
num = 1;
```



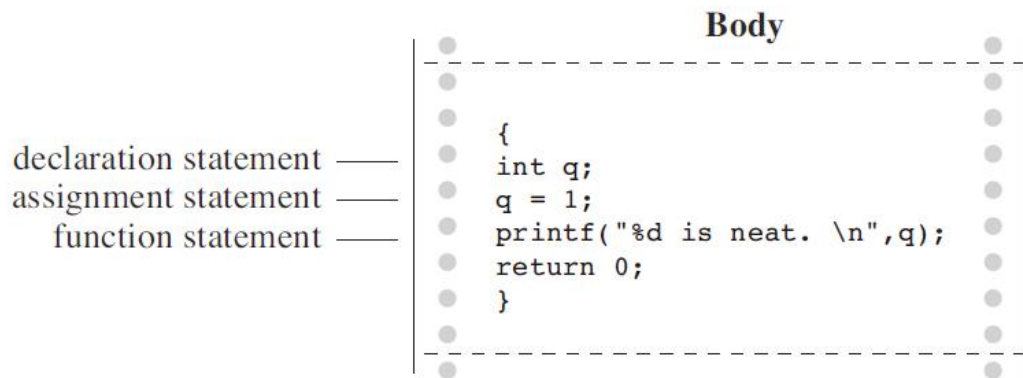
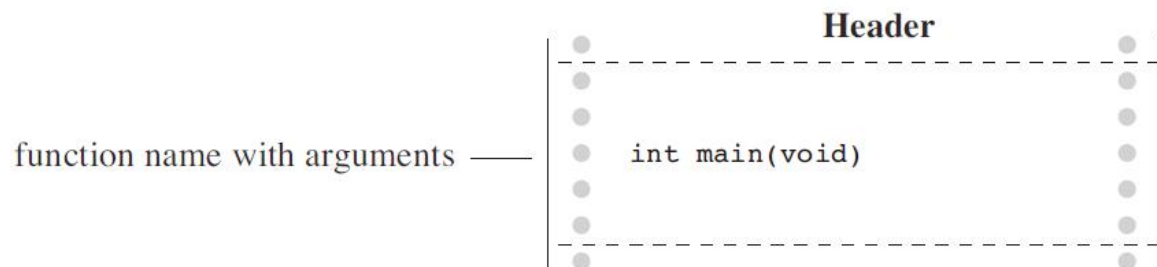
printf()函数:

```
printf( )
```

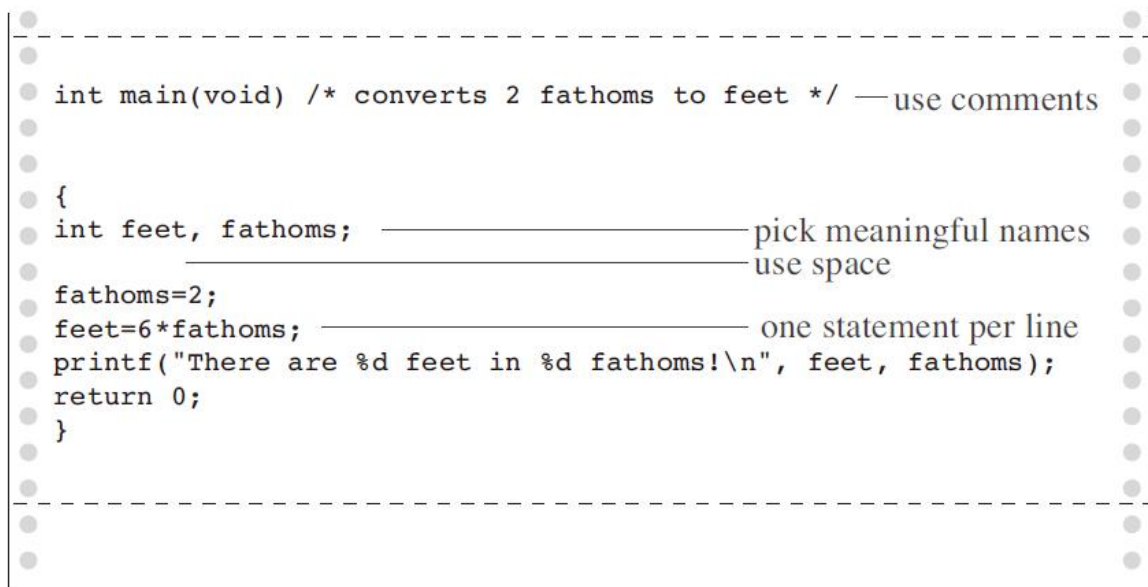


```
printf("That's mere contrariness!\n");
```

C语言程序结构



提升可读性:



- 写注释
- 变量名见词知意
- 缩进和空行