

C语言基本概念

张鲜玉

一个简单的C程序：

```
#include <stdio.h>
void main(){
    printf("Hello, Lvliang University.\n");
}
```

我们来梳理一下这个程序的内容结构：

```
指令
void main(){
    语句s
}
```

- 指令：是程序在预处理时要完成的操作，指令都以#开头
 - `#include` 表示包含其他文件的内容信息
 - `#include <stdio.h>` 表示本程序文件“包含”了C语言的标准输入/输出库文件（stdio.h）的信息。
 - 其他指令： `#define` `#if` `#endif` ...
 - 指令的结尾没有分号，且必须独占一行

- 函数：又名方法，是用来构建程序的结构块。一个C程序其实是函数的集合，函数分为两大类：一种是C语言所提供的标准库文件，是C语言的提供者已设计好的函数库，我们只需要引入这些文件就可以直接使用其中的函数；另一类是我们C语言的使用者自己写的函数（及函数集合）。
 - 术语“函数”来源于数学。在数学中，函数是指根据一个或多个给定参数进行一定运算的规则： $f(x) = x + 1$ 和 $g(y, z) = y^2 - z^2$ 。C语言的函数则更加宽泛，不一定必须有参数，也不一定要计算数值（其除了有运算的能力，还有执行其他命令的功能）。
 - 一个C程序可以包含很多函数，但只能有一个main函数，且必须有。main函数是所有程序文件的“主”函数，是程序执行的入口。

- 函数的结构：

- ```
返回类型 函数名(参数列表){
 声明
 语句
}
```

- 返回类型：是指每次调用此函数时，将返回的数据的类型

- C语言提供的数据类型如图所示

- 参数列表：是指要调用此函数时，需向函数提供的数据

- 函数名：是一个函数的标识符号

- 函数体：用花括号括起的部分是函数的函数体

- 函数体中，声明语句必须放在最前面，然后是其他语句（C99之后允许变量在使用之前声明即可）

- （更具体的讲解将在第n讲展开）

- 语句：是程序运行时执行的命令

- ```
#include <stdio.h>
void main(){
    printf("Hello, Lvliang University.\n");
}
```

- 语句的结尾都有分号。
- 在函数体中可以有0~多条语句，实例中我们只写了一条语句——printf()函数的调用语句。
- 这条函数调用语句是调用了C语言的标准输入/输出库文件的printf函数，所以要在程序的预处理指令中引入该文件。

- 注释：注释用于对代码的功能作用做阐释，帮助程序的阅读者理解代码。注释不会程序执行。
 - 多行注释： `/* */`
 - 单行注释： `//`

```
if(y!=0)
    if(x!=0)
        result = y/x;
else
    printf("Error");
```

- 这里的else子句属于内层if语句
- 如果想实现我们想要的效果，可以加上花括号

```
if(y!=0){
    if(x!=0)
        result = y/x;
}else
    printf("Error");
```

条件表达式

if语句使我们可以根据条件判断的结果来执行两个操作中的一个。

C语言还提供了一种特殊的运算符，允许我们根据条件判断来选择产生两个值中的一个值。