

# 编程

---

- 编程是编定程序的中文简称，就是让计算机代码解决某个问题，对某个计算体系规定一定的运算方式，使计算体系按照该计算方式运行，并最终得到相应结果的过程。
- 为了使计算机能够理解人的意图，人类就必须将需解决的问题的思路、方法和手段通过计算机能够理解的形式告诉计算机，使得计算机能够根据人的指令一步一步去工作，完成某种特定的任务。这种人和计算体系之间交流的过程就是编程。
- 编程：设计具备逻辑流动作用的一种“可控体系”，编程不一定是针对计算机程序而言的，针对具备逻辑计算力的体系，都可以算编程。



## 编程语言的种类

---

### 机器语言

---

在计算机系统中，一条机器指令规定了计算机系统的一个特定动作。一个系列的计算机在硬件设计制造时就用了若干指令规定了该系列计算机能够进行的基本操作，这些指令一起构成了该系列计算机的指令系统。在计算机应用的初期，程序员使用机器的指令系统来编写计算机应用程序，这种程序称为机器语言程序。使用机器语言编写的程序，由于每条指令都对应计算机一个特定的基本动作，所以程序占用内存少、执行效率高。缺点也很明显，如：编程工作量大，容易出错；依赖具体的计算机体系，因而程序的通用性、移植性都很差。

```
00 2007 # jump past sensor and constant locations
01 0000 # read right sensor value here
02 0000 # read left sensor value here
03 0000 # write right motor power level here
04 0000 # write left motor power level here
05 0000 # store motor-off constant here
06 0100 # store motor-on constant here
07 4011 # load right sensor value into register 1
08 4022 # load left sensor value into register 2
09 1123 # subtract 1 from 2 and store result in 3
10 4105 # load motor-off constant into register 1
11 4206 # load motor-on constant into register 2
12 3316 # if the left sensor is greater than the
13 5203 # right then turn the right motor on
14 5104 # and turn the left motor off
15 2007 # and then jump to beginning of the loop
16 5204 # else turn the left motor on
17 5103 # and turn the right motor off
18 2007 # and then jump to beginning of the loop
19 0000 # this location is not used by the program
```

## 汇编语言

---

为了解决使用机器语言编写应用程序所带来的一系列问题，人们首先想到使用助记符号来代替不容易记忆的机器指令。这种助记符号来表示计算机指令的语言称为符号语言，也称汇编语言。在汇编语言中，每一条用符号来表示的汇编指令与计算机机器指令一一对应；记忆难度大大减少了，不仅易于检查和修改程序错误，而且指令、数据的存放位置可以由计算机自动分配。用汇编语言编写的程序称为源程序，计算机不能直接识别和处理源程序，必须通过某种方法将它翻译成计算机能够理解并执行的机器语言，执行这个翻译工作的程序称为汇编程序。

```

-u cs:10
076A:0010 BB0000      MOV     BX,0000
076A:0013 B80000      MOV     AX,0000
076A:0016 B90800      MOV     CX,0008
076A:0019 2E          CS:
076A:001A 0307          ADD     AX,[BX]
076A:001C 83C302          ADD     BX,+02
076A:001F E2F8          LOOP    0019
076A:0021 B8004C          MOV     AX,4C00
076A:0024 CD21          INT     21
076A:0026 E89F0E          CALL    0EC8
076A:0029 83C404          ADD     SP,+04
076A:002C 3DFFFF          CMP     AX,FFFF
076A:002F 7403          JZ      0034

```

## 汇编程序

汇编程序。使用汇编语言编写计算机程序，程序员仍然需要十分熟悉计算机系统的硬件结构，所以从程序设计本身上来看仍然是低效率的、繁琐的。但正是由于汇编语言与计算机硬件系统关系密切，在某些特定的场合，如对时空效率要求很高的系统核心程序以及实时控制程序等，迄今为止汇编语言仍然是十分有效的程序设计工具。但它有不可替代的特性，比如一些单片机或者一些直接控制硬件的程序就一定要用汇编语言

```

data_seg1 segment
..... <在此进行数据存储单元的定义和分配等伪操作>
data_seg1 ends
data_seg2 segment
..... <在此进行数据存储单元的定义和分配等伪操作>
data_seg2 ends
code_seg segment
    assume cs:code_seg, ds:data_seg1, es:data_seg2
start:
    mov     ax, data_seg1
    mov     ds, ax
    mov     ax, data_seg2
    mov     es, ax
    ..... <在此填加程序代码>
code_seg ends
end start

```

## 高级语言

- 高级语言是一类接近于人类的自然语言和数学语言的程序设计语言的统称。按照其程序设计的出发点和方式不同，高级语言分为了面向过程的语言和面向对象的语言，如Fortran语言、C语言、汉语程序设计语言等都是面向过程的语言；而以C++、Smalltalk语言等为代表的面向对象的语言与面向过程语言有着许多不同，这些语言支持“程序是相互联系的离散对象集合”，这样一种新的程序设计思维方式，具有封装性、继承性和多态性等特征。
- 高级语言按照一定的语法规则，由表达各种意义的运算对象和运算方法构成。使用高级语言编写程序的优点是：编程相对简单、直观、易理解、不容易出错；高级语言是独立于计算机的，因而用高级语言编写的计算机程序通用性好，具有较好的移植性。用高级语言编写的程序称为源程序，计算机系统不能直接理解和执行，必须通过一个语言处理系统将其转换为计算机系统能够认识、理解的目标程序才能被计算机系统执行。



```

1  #define M_TAU (2*M_PI)
2  float fast_cosc(float x){
3      x -= 0.25f + floorf(x + 0.25f);
4      x *= 16.0f * (fabs(x) - 0.5f);
5      x += 0.225f * x * (fabs(x) - 1.0f);
6      return x;
7  }
8  float fast_sinf(float x){
9      return fast_cosc(x/M_TAU - 0.25f);
10 }
11 float fast_cosf(float x){
12     return fast_cosc(x/M_TAU);
13 }

```

## 编程语言的执行原理

- 计算机对除机器语言以外的源程序不能直接识别、理解和执行，都必须通过某种方式转换为计算机能够直接执行的。
- 这种将高级程序设计语言编写的源程序转换到机器目标程序的方式有两种：解释方式和编译方式。
- 解释方式下，计算机对高级语言书写的源程序一边解释一边执行，不能形成目标文件和执行文件。
- 编译方式下，首先通过一个对应于所用程序设计语言的编译程序对源程序进行处理，经过对源程序的词法分析、语法分析、语意分析、代码生成和代码优化等阶段将所处理的源程序转换为用二进制代码表示的目标程序，然后通过连接程序处理将程序中所用的函数调用、系统功能调用等嵌入到目标程序中，构成一个可以连续执行的二进制执行文件。调用这个执行文件就可以实现程序员在对应源程序文件中所指定的相应功能。

