

计算机理论基础

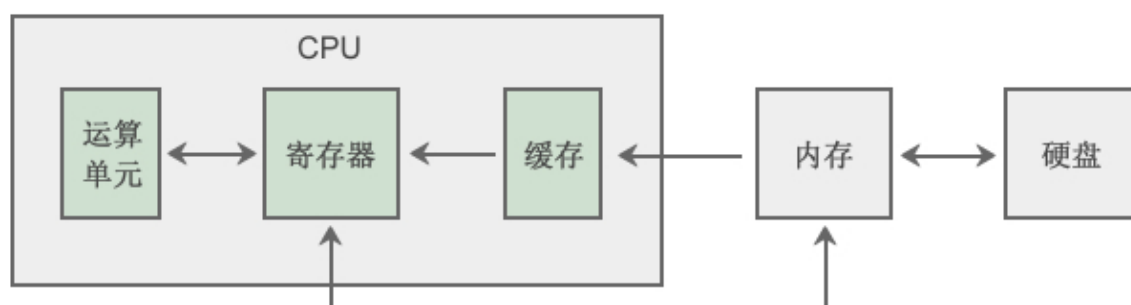
1.1、一个程序在计算机中是如何运行的？(了解)

1.1.1、核心部件

程序是保存在硬盘中的，要载入内存才能运行，CPU也被设计为只能从内存中读取数据和指令；

对于CPU来说，内存仅仅是一个存放指令和数据的地方，并不能在内存中完成计算功能，为了了解具体的运算过程，先看一下CPU的结构：

CPU是一个复杂的计算机部件，它内部又包含很多小零件，如下图：



#1 运算单元是CPU的大脑，负责加减乘除、比较、位移等运算工作，每种运算都有对应的电路支持，速度很快。

寄存器是CPU内部非常小、非常快速的存储部件，它的容量有限，对于32位的CPU，每个寄存器一般能存储**32位（4个字节）**的数据，对于64位的CPU，每个寄存器一般能存储64位（8个字节）的数据。为了完成各种复杂的功能，现代CPU都内置了几十个甚至上百个寄存器，嵌入式系统功能单一，寄存器数量较少；

我们经常所说的多少位CPU，指的是寄存器的位数；寄存器在程序的执行过程中至关重要，不可或缺，它们可以用来完成数学运算、控制循环次数、控制程序的执行流程、标记CPU的运行状态等。

#2 CPU内部为什么又要设置缓存呢？

虽然内存的读取速度已经很快了，但是和CPU相比，还是有很大差距的，不是一个数量级的，如果每次都从内存中读取数据，会严重拖慢CPU的运行速度，CPU经常处于等待状态，无事可做。在CPU内部设置一个缓存，可以将使用频繁的数据暂时读取到缓存，需要同一地址上的数据时，就不用大老远的再去访问内存，直接从缓存中读取即可。

大家在购买CPU时也会经常关心缓存容量，例如：Intel Core i7 3770K 的三级缓存为8MB，二级缓存为256KB，一级缓存为32KB，容量越大，CPU越强悍。

#3 缓存命中率

缓存的容量是有限的，CPU只能从缓存中读取到部分数据，对于使用不是很频繁的数据，会绕过缓存，直接到内存中读取。所以不是每次都能从缓存中得到数据，这就是缓存的命中率，能够从缓存中读取就命中，否则就没命中。关于缓存的命中率又是一门学问，哪些数据保留在缓存，哪些数据不保留，都有复杂的算法。

#4 CPU 指令集

要想让CPU工作，必须借助特定的指令，例如 add 用于加法运算，sub 用于减法运算，cmp 用于比较两个数的大小，这称为CPU的指令集（Instruction Set）。我们的C语言代码最终也会编译成一条一条的CPU指令。不同型号的CPU支持的指令集会有所差异，但绝大部分是相同的。

```
// 需求：完成1+1的功能
mov 寄存器A 1;
add 寄存器A 1;
```

实际上，上面的代码是汇编语言，不是CPU指令，汇编语言还要经过简单的转换才能成为CPU指令；为了更加容易地说明问题，这些语句也没有严格遵守汇编的语法。有兴趣的同学可以自行学习汇编语言，这里不再展开讲解。

1.1.2、载入内存，让程序运行起来

如果你的电脑上安装了QQ，你希望和好友聊天，会双击QQ图标，打开QQ软件，输入账号和密码，然后登录就可以了。那么，QQ是怎么运行起来的呢？

首先，有一点你要明确，你安装的QQ软件是保存在硬盘中的。双击QQ图标，操作系统就会知道你要运行这个软件，它会在硬盘中找到你安装的QQ软件，将数据（安装的软件本质上就是很多数据的集合）复制到内存。对！就是复制到内存！QQ不是在硬盘中运行的，而是在内存中运行的。为什么呢？因为内存的读写速度比硬盘快很多。对于读写速度，**内存 > 固态硬盘 > 机械硬盘**。机械硬盘是**靠电机带动盘片转动来读写数据**的，而内存条通过电路来读写数据，电机的转速肯定没有电的传输速度（几乎是光速）快。虽然固态硬盘也是**通过电路来读写数据**，但是因为与内存的控制方式不一样，速度也不及内存。所以，不管是运行QQ还是编辑Word文档，都是先将硬盘上的数据复制到内存，才能让CPU来处理，这个过程就叫作载入内存（Load into Memory）。完成这个过程需要一个特殊的程序（软件），这个程序就叫做加载器（Loader）。

CPU直接与内存打交道，它会读取内存中的数据进行处理，并将结果保存到内存。如果需要保存到硬盘，才会将内存中的数据复制到硬盘。例如，打开Word文档，输入一些文字，虽然我们看到的不一樣了，但是硬盘中的文档没有改变，新增的文字暂时保存到了内存，**Ctrl+S**才会保存到硬盘。因为内存断电后会丢失数据，所以如果你编辑完Word文档忘记保存就关机了，那么你将永远无法找回这些内容。

#1 虚拟内存

如果我们运行的程序较多，占用的空间就会超过内存（内存条）容量。例如计算机的内存容量为2G，却运行着10个程序，这10个程序共占用3G的空间，也就意味着需要从硬盘复制3G的数据到内存，这显然是不可能的。操作系统（Operating System，简称OS）为我们解决了这个问题：当程序运行需要的空间大于内存容量时，会将内存中暂时不用的数据再写回硬盘；需要这些数据时再从硬盘中读取，并将另外一部分不用的数据写入硬盘。这样，硬盘中就会有一部分空间用来存放内存中暂时不用的数据。这一部分空间就叫做虚拟内存（Virtual Memory）。 $3G - 2G = 1G$ ，上面的情况需要在硬盘上分配1G的虚拟内存。硬盘的读写速度比内存慢很多，反复交换数据会消耗很多时间，所以如果你的内存太小，会严重影响计算机的运行速度，甚至会出现“卡死”现象，即使CPU强劲，也不会有大的改观。如果经济条件允许，建议将内存升级为至少8G，在win7、win10下运行软件就会比较流畅了。

思考题

1. 计算机中哪一个硬件设备负责执行程序？

CPU

2. 内存 的速度快还是 硬盘 的速度快？

内存

3. 我们的程序是安装在内存中的，还是安装在硬盘中的？

硬盘

4. 我买了一个内存条，有 500G 的空间!!!，这句话对吗？

不对，内存条通常只有 4G / 8G / 16G / 32G

5. 计算机关机之后，内存中的数据都会消失，这句话对吗？

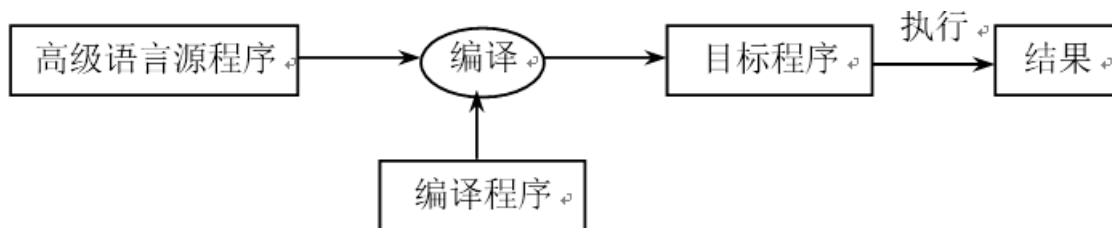
正确

1.1.3、程序(Java)的执行流程

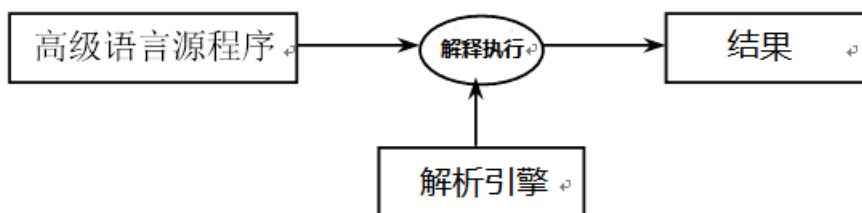
语言处理程序的功能是将除机器语言以外，利用其他计算机语言编写的程序，转换成机器所能直接识别并执行的机器语言程序的程序。可以分为三种类型，即汇编程序、**编译程序**和**解释程序**。

通常将汇编语言及各种高级语言编写的计算机程序称为**源程序 (Source Program)**，而把由源程序经过翻译（汇编或者编译）而生成的机器指令程序称为**目标程序 (Object Program)**。语言处理程序中的汇编程序与编译程序具有一个共同的特点，即**必须生成的目标程序**，然后通过执行目标程序得到最终结果。

编译程序 (Compiler, compiling program)，也称为编译器，是指把用高级程序设计语言书写的源程序，转换成等价的机器语言格式的目标程序。



解释程序 (解析引擎)是对源程序进行解释（逐句翻译），翻译一句执行一句，边解释边执行，从而得到最终结果。解释程序不产生将被执行的目标程序，而是借助解释程序直接执行源程序本身。



Java是一种新型的**跨平台分布式**和程序设计语言。Java以它简单、安全、可移植、面向对象、多线程处理和具有动态等特性引起世界范围的广泛关注。Java语言是基于C++的，其最大的特色在于“**一次编译，处处运行**”。Java已逐渐成为网络化软件的核心语言。

1.1.4、计算机采用二进制原因(理解)

计算机只认识0和1，计算机的世界是二进制世界。人类是十进制世界。

二进制 => 十进制

十进制 => 二进制

结论：计算机世界和人类世界可以相互转换。

首先，二进位计数制仅用两个数码。0和1，所以，任何具有二个不同稳定状态的元件都可用来表示数的某一位。而在实际上具有两种明显稳定状态的元件很多。例如，氖灯的“亮”和“熄”；开关的“开”和“关”；电压的“高”和“低”、“正”和“负”；纸带上的“有孔”和“无孔”，电路中的“有信号”和“无信号”，磁性材料的南极和北极等等，不胜枚举。利用这些截然不同的状态来代表数字，是很容易实现的。不仅如此，更重要的是两种截然不同的状态不单有量上的差别，而且是有质上的不同。这样就能大大提高机器的抗干扰能力，提高可靠性。而要找出一个能表示多于二种状态而且简单可靠的器件，就困难得多了。

给定1个二进制位，可以存储2种状态(0 / 1)
给定2个二进制位，可以存储4种状态(00 / 01 / 10 / 11)
给定3个二进制位，可以存储8种状态(000 / 001 / 010 / 011 / 100 / 101 / 110 / 111)
...
n 2的n次方个状态。

总结：
给定n个二进制位，可以表示2的n次方种状态。

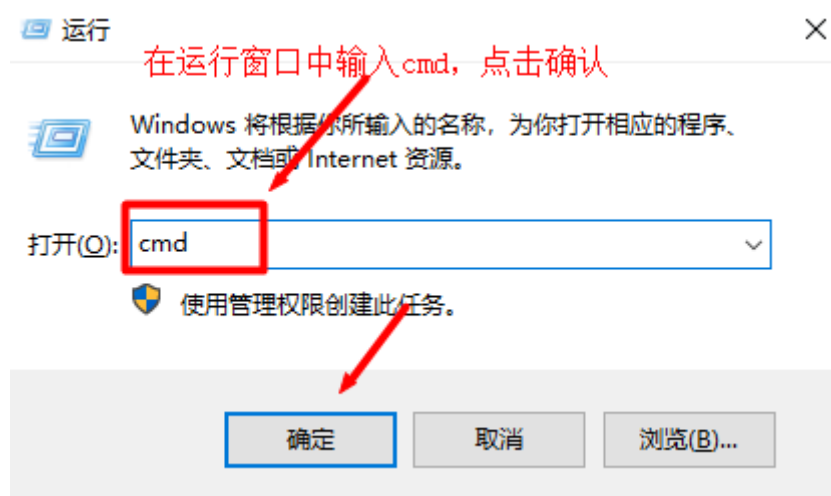
其次，二进位计数制的四则运算规则十分简单。而且四则运算最后都可归结为加法运算和移位，这样，电子计算机中的运算器线路也变得十分简单了。不仅如此，线路简化了，速度也就可以提高。这也是十进位计数制所不能相比的。

第三，在电子计算机中采用二进制表示数可以节省设备。可以从理论上证明，用三进位制最省设备，其次就是二进制。但由于二进制有包括三进位制在内的其他进位制所没有的优点，所以大多数电子计算机还是采用二进制。此外，由于二进制中只用二个符号“0”和“1”，因而可用布尔代数来分析和综合机器中的逻辑线路。这为设计电子计算机线路提供了一个很有用的工具。

1.2、windows常见操作（掌握）

1.2.1、系统中DOS的基本操作

操作一：打开dos窗口，在电脑键盘上点击按中win+R，打开电脑的运行命令，在运行的框中输入cmd，操作图片步骤如下。



常见技巧一：dos命令窗口打开之后，系统会自动打开【C:\Users\Administrator>】这个时候可以通过盘符加上：切换到其他磁盘，操作图片步骤如下。

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.17763.1217]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>f:

F:\>
```

输入f:直接切换到f盘

常见技巧二：在dos命令窗口中查看当前磁盘中的目录，可以在输入dir查看目录，操作图片步骤如下。

```
F:\>dir

F:\ 的目录
                输入dir查询f盘的目录
2015/12/27 周日 18:25    <DIR>          apache-tomcat-7.0.57
2017/12/13 周三 23:37    <DIR>          api
2016/11/28 周一 23:23    <DIR>          DTD
2016/01/31 周日 14:37    21,358,715     hapu_最新.rar
2016/11/20 周日 21:11    <DIR>          hibernate-release-4.2.0.Final
2020/05/27 周三 22:10    <DIR>          IdeaProjects
2016/10/12 周三 10:41    <DIR>          jars
2020/05/14 周四 08:22    <DIR>          Java
2018/05/05 周六 14:09    <DIR>          JavaPros
2018/05/07 周一 22:08    <DIR>          JavaTools
2017/06/12 周一 16:24    <DIR>          MavenPros
2018/06/22 周五 23:52    <DIR>          MyRepository
```

常见技巧三：进入单级进入文件夹cd + 文件名字，多级文件夹cd + 文件夹名字\文件夹名字\...操作图片步骤如下。

```
### 进入test目录
cd test

### 进入test目录下的java目录
cd test/java

### 表示退到当前目录的上级目录
cd ..

### 表示进入当前盘符的根目录
cd /
```

```
F:\>cd Java

F:\Java>
```

要求：准备开发目录！

- 1> 准备开发盘。
- 2> 找一个目录作为开发目录。

1.2.2、常用快捷方式

1. **Win + D**：该组合键会快速返回桌面并将所有的界面最小化，再次按会还原会之前的桌面状态。
2. **Alt + Tab**：经典的窗口切换组合键，它为每个窗口提供了缩略图预览，继续按快捷键切换窗口，然后在高亮显示所需窗口时释放组合键。
3. **Win + 方向键**：将窗口快速捕捉到屏幕的任何角落，向左和向右将窗口捕捉到屏幕的这些侧面，而向上和向下将窗口捕捉到角落。按Win + 上或下两次将使一个窗口全屏显示或最小化。
4. **Ctrl + A** (命令提示符中有效)：快速全选当前所有文本。
5. **Ctrl + C** (命令提示符中有效) – 与以前的版本不同，您可以简单地使用普通键盘Ctrl + C来复制命令提示符中的文本或输出。或者，也可以使用快捷键Ctrl + Insert。
6. **Ctrl + V** (命令提示符中有效)：就像复制操作一样，您可以使用常规快捷键Ctrl + V将文本或命令粘贴到命令提示符中。或者，您也可以使用快捷键Shift + Insert。
7. **Win+E** 打开Windows资源管理器Explorer【即我的电脑、计算机】
8. **Win+L** 锁定计算机或切换用户

1.3、编程语言（了解）

1.3.1、什么是编程？什么是编程语言？

编程是个动词，编程==写代码，写代码为了什么？为了让计算机干你想要做的事情，比如，马化腾想跟别人聊天，于是写了个聊天软件，这个软件就是一堆代码的集合，这些代码是什么？这些代码是计算机能理解的语言。

那计算机能理解的语言是什么呢？之前，我们已经了解到，它只能理解2进制，0101010...，你总不能人肉输一堆二进制给计算机(虽然最原始的计算机就是这么干的)让它工作吧，这样开发速度太慢了。所以最好的办法就是人输入简单的指令，计算机能把指令转成二进制进行执行，举例如下：

假如 程序员想让计算机 播放一首 歌曲，只需要输入指令，

1. open "老男孩.mp3"
2. play

计算机的CPU接收到这样的指令后，会把它转成一堆 只有cpu可以理解的指令，然后再将指令变成各种对应的如下类似二进制

1. [op | rs | rt | address/immediate]
2. 35 3 8 68 decimal
3. 100011 00011 01000 00000 00001 000100 binary

最终cpu 去调用你的硬盘上这首歌，通过音箱播放。

上面cpu那段指令太难理解了，如果让你天天写这样的代码，大家非得自杀不可。还好，伟大的计算机先驱们，开发了各种编程语言，让我们只需要通过写一些简单的规则，就能操作计算机工作啦。

1.3.2、常见的编程语言

机器语言

优点是最底层，速度最快，缺点是最复杂，开发效率最低

汇编语言

优点是比较底层，速度最快，缺点是复杂，开发效率最低

高级语言

编译型语言执行速度快，不依赖语言环境运行，跨平台差

解释型跨平台好，一份代码，到处使用，缺点是执行速度慢，依赖解释器运行

Jan 2020	Jan 2019	Change	Programming Language	Ratings	Change
1	1		Java	16.896%	-0.01%
2	2		C	15.773%	+2.44%
3	3		Python	9.704%	+1.41%
4	4		C++	5.574%	-2.58%
5	7	⬆	C#	5.349%	+2.07%
6	5	⬇	Visual Basic .NET	5.287%	-1.17%
7	6	⬇	JavaScript	2.451%	-0.85%
8	8		PHP	2.405%	-0.28%
9	15	⬆	Swift	1.795%	+0.61%
10	9	⬇	SQL	1.504%	-0.77%
11	18	⬆	Ruby	1.063%	-0.03%
12	17	⬆	Delphi/Object Pascal	0.997%	-0.10%
13	10	⬇	Objective-C	0.929%	-0.85%
14	16	⬆	Go	0.900%	-0.22%
15	14	⬇	Assembly language	0.877%	-0.32%
16	20	⬆	Visual Basic	0.831%	-0.20%
17	25	⬆	D	0.825%	+0.25%
18	12	⬇	R	0.808%	-0.52%
19	13	⬇	Perl	0.746%	-0.48%
20	11	⬇	MATLAB	0.737%	-0.76%

比尔·盖茨亲自授课--计算机原理

<https://www.bilibili.com/video/BV1Lb411J7oq?p=1>