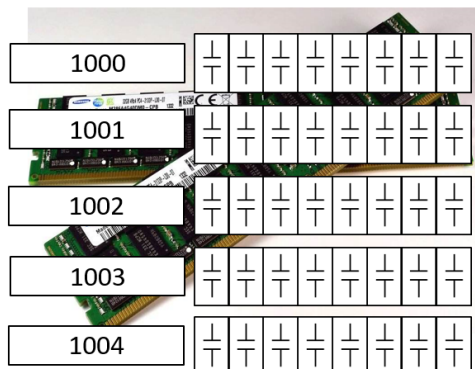


1、内存

存储器

保存正在运行的程序指令和数据



- 关机后，内存里的指令和数据全部丢失。
- 内存上的一个最小物理器件只能存储一个位（二进制），所以它的名称也叫做位（bit，缩写为b）。一般8个位组成一个字节（Byte，缩写为B），若干个字节组成一个字（word）。
- 在一般的机器中，**内存按字节编址**，内存大小也是按字节计量。

内容定位：

很重要，对以后的学习有很大影响。

传说中C++最难的部分——**指针**的本质就是内存地址。

掌握程度：

建议现在掌握：**理解并记住它们。**

内容重点：

首先分清楚两个概念：**内存空间**和**内存地址**

内存空间是用来存储数据或指令的，我们买电脑的时候说的内存条的大小（4G、8G等等）就是指内存空间的大小。它是用来放东西的，所以它的大小会影响电脑性能。

计算机领域常见的表示信息量大小的单位及换算关系：

GB，生活中一般简称G，比如4G内存、512G硬盘等，就是说存储空间大小为4GB、512GB。

MB，生活中一般简称M，也可读写为“兆”，比如CodeBlocks的安装包文件大小为164M。

KB，生活中一般简称K，比如四六级考试的照片要求小于200KB（理论上这里的B必须是大写，但是很多人用的不规范，所以你们会见到各种大小写混用的情况，但是你们要明白，KB才是规范的写法）。

Byte，字节，生活中可以简称为B，计算机领域只能用大写的B来简写。

bit，位，计算机领域只能用小写的b来简写。

1GB=1024MB，1MB=1024KB，1KB=1024B，1B=8bit。

内存地址是内存空间中每个字节的门牌号，仅仅是一个抽象的编号。计算机系统的位数不同，会影响计算机分配的内存地址大小。

比如，32位系统最大只能给内存分配32位地址，也就是门牌号最大到二进制的32个1，十进制的 $2^{32}-1$ ，每个门牌号下可以放1字节（即8位）的内容，所以系统能够编址并使用的内存空间大小就是4G，就算插了一个8G的内存条，也只能用4G。

（有同学问：剩下的内存还可以用吗？很可惜，不能。在电脑看来，没有地址的内存块就像你们玩游戏时没有解锁的地图，不知道前面是什么，无法识别，就无法操作，那么在当前的系统下，自然就无法使用。）

然后我们需要记一些相关基本知识。

内存的最基本组成颗粒叫作“位”（表示数据大小的那个位，不是系统位数的位）。

它的物理实体是一个电容，有已充电和未充电两种状态，分别表示1和0。

所以，一个位可以存储一个二进制数字，它的值只能是0或1。

每8个相邻的“位”组成一个“字节”，它可以存储一个八位二进制数。

（八位二进制数的范围是从00000000到11111111，即十进制的0到255。）

内存空间中的每个字节（**注意是字节**）都有一个属于它的地址，称为内存地址。一个位是没有专属于它的地址的。

内存地址是连续的（**从0开始往上挨着数**）。

每个地址对应的1个字节的内存空间自然也是连续的。

在本门课中，内存地址默认为32位。（从长远意义看，这种预设下写出来的程序可以用在32位和64位的系统里，但是如果默认是64位，就不能用在32位系统里了。从现实意义看，我们这门课绝对不可能需要写一个需要4G内存以上才能运行的程序。）

最后，让我们再次确认一下，你真的搞清楚内存地址和内存空间了吗？

下图表示了内存中的一个字节，它所占用的内存空间是8位，而它的内存地址一共有32位，是“000...01111101100”（开头22个零），十进制表示为1004。



那么，因为内存地址是连续的，所以下一个字节的地址就是1005。

如果你已经搞懂了，恭喜你，本门课程最最最难的指针，你已经打好了坚实的第一步理论基础！

2、控制器解析指令

内容定位：

下图是课本中本章节没有提及的内容。

这部分内容将有助于理解计算机的**运行原理**。

与本门课程关系较小，但对计算机的专业的核心课程：**计算机组成原理**有较大帮助。

掌握程度：

对计算机专业有兴趣的同学，理解它们。

对计算机专业兴趣一般的同学，了解即可。

控制信号

控制器将指令解码后得到操作码、寄存器编号、**内存地址**等与执行指令有关的信息，从而生成相应的控制信号发送给其它设备。

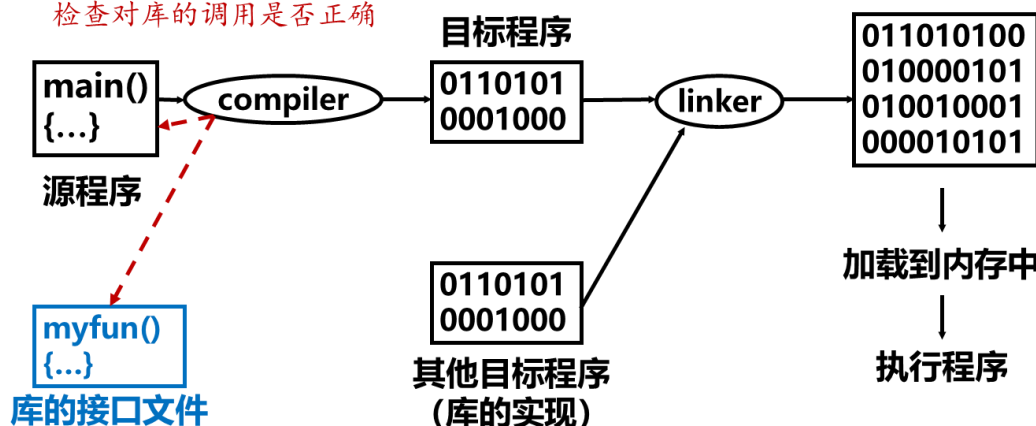
例：将内存地址为10086的数据加1，再显示该结果。

指令 1	操作：读取	运算器R4	内存地址：10086	
指令 2	操作：相加	运算器R4	运算器R4	常数：1
指令 3	操作：写入	运算器R4	内存地址：10086	
指令 4	操作：输出	显示器R13	内存地址：10086	

3、程序编译与执行过程

程序编译与执行过程

在编译时，首先根据库包含
检查对库的调用是否正确



内容定位：

上图是课本中1.4.3节的内容以及相关的扩展知识。

在本学期后续章节的学习中，我们还会多次提起“编译时”、“运(执)行时”，表示程序编译与执行过程中的不同阶段。

掌握程度：

建议现在掌握：**理解**它们。

内容重点：

C++源程序在预编译后成为“完整的源程序”，再经正式编译后成为可运行的机器码程序。因此，这三个主要阶段的顺序是“预编译”->“正式编译”->“运行”。

二、各类补充知识（与本课程考查范围无关）

1、编码的发展史和不同字符集编码规则：

<https://blog.csdn.net/yaomingyang/article/details/79374209>

2、程序员买苹果之编程到底难在哪里？

<https://www.zhihu.com/question/22508677/answer/141334678>