

b中有任何不能访问的文件，则是因为/b指向了CD-ROM的根目录。（在开始时，不能访问这些文件似乎并不是一个严重问题：文件系统几乎总是安装在空目录上。）如果系统有多个硬盘，它们也可以都安装在单个树上。

在UNIX中，另一个重要的概念是特殊文件（special file）。提供特殊文件是为了使I/O设备看起来像文件一般。这样，就像使用系统调用读写文件一样，I/O设备也可通过同样的系统调用进行读写。有两类特殊文件：块特殊文件（block special file）和字符特殊文件（character special file）。块特殊文件指那些由可随机存取的块组成的设备，如磁盘等。比如打开一个块特殊文件，然后读第4块，程序可以直接访问设备的第4块而不必考虑存放该文件的文件系统结构。类似地，字符特殊文件用于打印机、调制解调器和其他接收或输出字符流的设备。按照惯例，特殊文件保存在/dev目录中。例如，/dev/lp是打印机（曾经称为行式打印机）。

在本节中讨论的最后一个特性既与进程有关也与文件有关：管道。管道（pipe）是一种虚文件，它可连接两个进程，如图1-16所示。如果进程A和B希望通过管道对话，它们必须提前设置该管道。当进程A想对进程B发送数据时，它把数据写到管道上，仿佛管道就是输出文件一样。进程B可以通过读该管道而得到数据，仿佛该管道就是一个输入文件一样。这样，在UNIX中两个进程之间的通信就很类似于普通文件的读写了。更为强大的是，若进程要想发现它所写入的输出文件不是真正的文件而是管道，则需要使用特殊的系统调用。文件系统是非常重要的。我们将在第6章，以及第10章和第11章中具体讨论它们。



图1-16 由管道连接的两个进程

1.5.4 输入/输出

所有的计算机都有用来获取输入和产生输出的物理设备。毕竟，如果用户不能告诉计算机该做什么，而在计算机完成了所要求的工作之后竟不能得到结果，那么计算机还有什么用处呢？有各种类型的输入和输出设备，包括键盘、显示器、打印机等。对这些设备的管理全然依靠操作系统。

所以，每个操作系统都有管理其I/O设备的I/O子系统。某些I/O软件是设备独立的，即这些I/O软件部分可以同样应用于许多或者全部的I/O设备上。I/O软件的其他部分，如设备驱动程序，是专门为特定的I/O设备设计的。在第5章中，我们将讨论I/O软件。

1.5.5 保护

计算机中有大量的信息，用户经常希望对其进行保护，并保守秘密。这些信息可包括电子邮件、商业计划、退税等诸多内容。管理系统的安全性完全依靠操作系统，例如，文件仅供授权用户访问。

作为一个简单的例子，以便读者对如何实现安全有一个概念，请考察UNIX。UNIX操作系统通过对每个文件赋予一个9位的二进制保护代码，对UNIX中的文件实现保护。该保护代码有三个3位字段，一个用于所有者，一个用于所有者同组（用户被系统管理员划分成组）中的其他成员，而另一个用于其他人。每个字段中有一位用于读访问，一位用于写访问，一位用于执行访问。这些位就是知名的rwx位。例如，保护代码rwxr-x--x的含义是所有者可以读、写或执行该文件，其他的组成员可以读或执行（但不能写）该文件，而其他人可以执行（但不能读和写）该文件。对一个目录而言，x的含义是允许查询。一条短横线的含义是，不存在对应的许可。

除了文件保护之外，还有很多有关安全的问题存在。保护系统不被人类或非人类（如病毒）入侵，则是其中之一。我们将在第9章中研究各种安全性问题。

1.5.6 shell

操作系统是进行系统调用的代码。编辑器、编译器、汇编程序、链接程序以及命令解释器等，尽管非常重要，也非常有用，但是它们确实不是操作系统的组成部分。为了避免可能发生的混淆，本节将大致介绍一下UNIX的命令解释器，称为shell。尽管shell本身不是操作系统的一部分，但它体现了许多操作系统的特性，并很好地说明了系统调用的具体用法。shell同时也是终端用户与操作系统之间的界面，除非用户使用的是一个图形用户界面。有许多种类的shell，如sh、csh、ksh以及bash等。它们全部支持下面所介绍的功能，这些功能可追溯到早期的shell（即sh）。