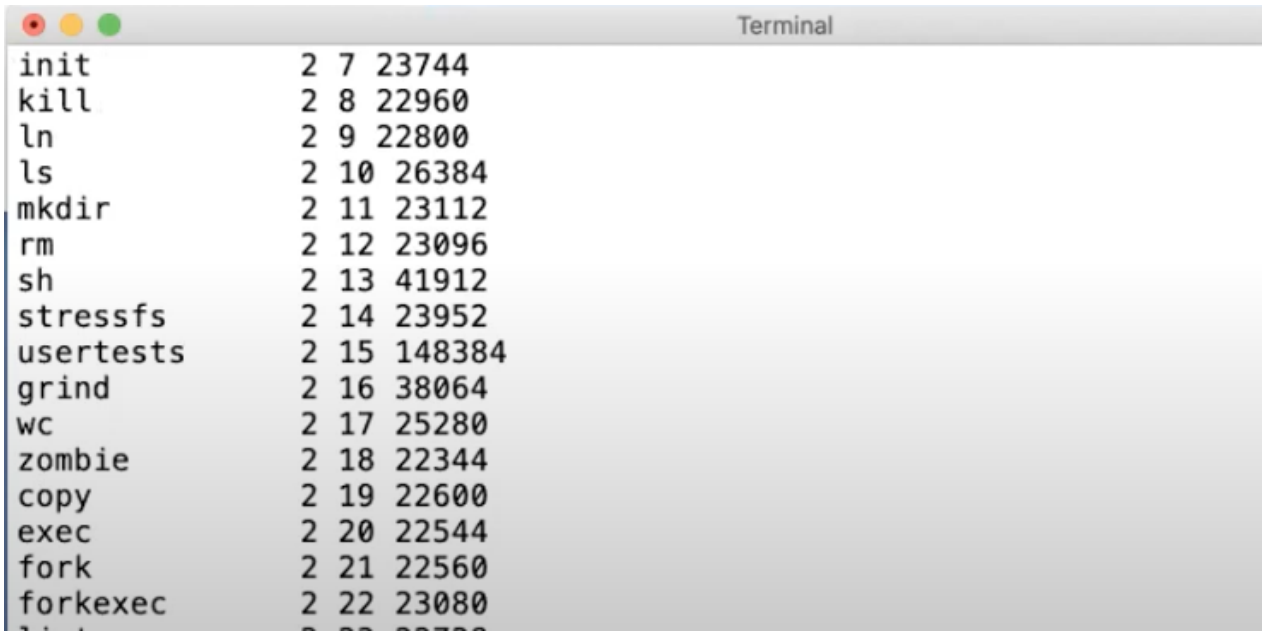
1.7 Shell

我说了很多XV6的长的很像Unix的Shell，相对图形化用户接口来说，这里的Shell通常也是人们说的命令行接口。如果你还没有用过Shell，Shell是一种对于Unix系统管理来说非常有用的接口，它提供了很多工具来管理文件，编写程序，编写脚本。你之前看过我演示一些Shell的功能，通常来说，当你输入内容时，你是在告诉Shell运行相应的程序。所以当我输入ls时，实际的意义是我要求Shell运行名为ls的程序，文件系统中会有一个文件名为ls，这个文件中包含了一些计算机指令，所以实际上，当我输入ls时，我是在要求Shell运行位于文件ls内的这些计算机指令。

我现在运行ls，它实际的工作就是输出当前目录的文件列表。



你可以看到第4行，就是一个叫做ls的文件。这就是包含了计算机指令的文件。

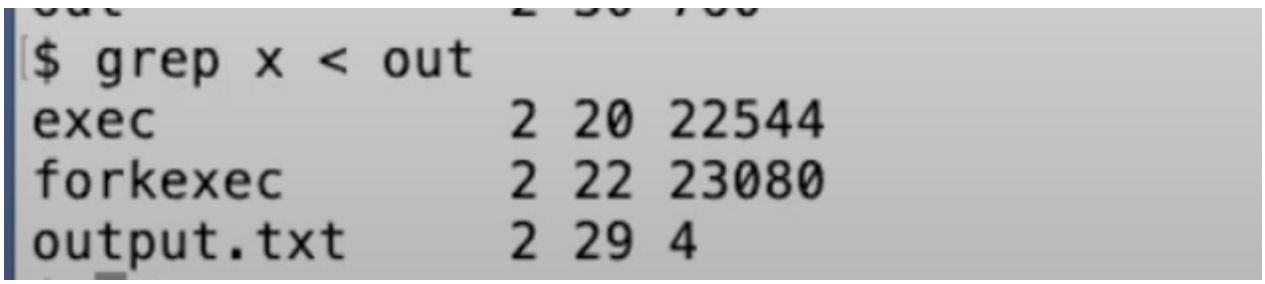
除了运行程序以外，Shell还会做一些其他的事情，比如，它允许你能重定向IO。比如，我输入 ls > out

这里的实际意义是，我要求Shell运行ls命令，但是将输出重定向到一个叫做out的文件中。这里执行完成之后我们看不到任何的输出，因为输出都送到了out文件。现在我们知道out文件包含了一些数据，我们可以通过cat指令读取一个文件，并显示文件的内容。

之后我们可以看到ls指令相同的输出。

你也可以运行一个名为grep的指令，并将x作为参数传给grep。

grep x会搜索输入中包含x的行，我可以告诉shell将输入重定向到文件out，这样我们就可以查看out中的x。



因为out文件包含了ls的输出，所以我们可以看出有3个文件的文件名包含了x。

我们将会花很多时间在Shell上，Shell是最传统，最基础的Unix接口。因为当Unix最开始被开发出来时，只有简单的终端接口，例如Shell。Unix最早的用途是给多个用户分时复用机器，用户通过Shell与机器交互。

> 学生提问：有一个系统调用和编译器的问题。编译器如何处理系统调用？生成的汇编语言是不是会调用一些由操作系统定义的代码段?
>
> Robert教授：有一个特殊的RISC-V指令，程序可以调用这个指令，并将控制权交给内核。所以，实际上当你运行C语言并执行例如open或者write的系统调用时，从技术上来说，open是一个C函数，但是这个函数内的指令实际上是机器指令，也就是说我们调用的open函数并不是一个C语言函数，它是由汇编语言实现，组成这个系统调用的汇编语言实际上在RISC-V中被称为ecall。这个特殊的指令将控制权转给内核。之后内核检查进程的内存和寄存器，并确定相应的参数。