重定向 管道 C 语言编写

部分课件来自于李中国老师

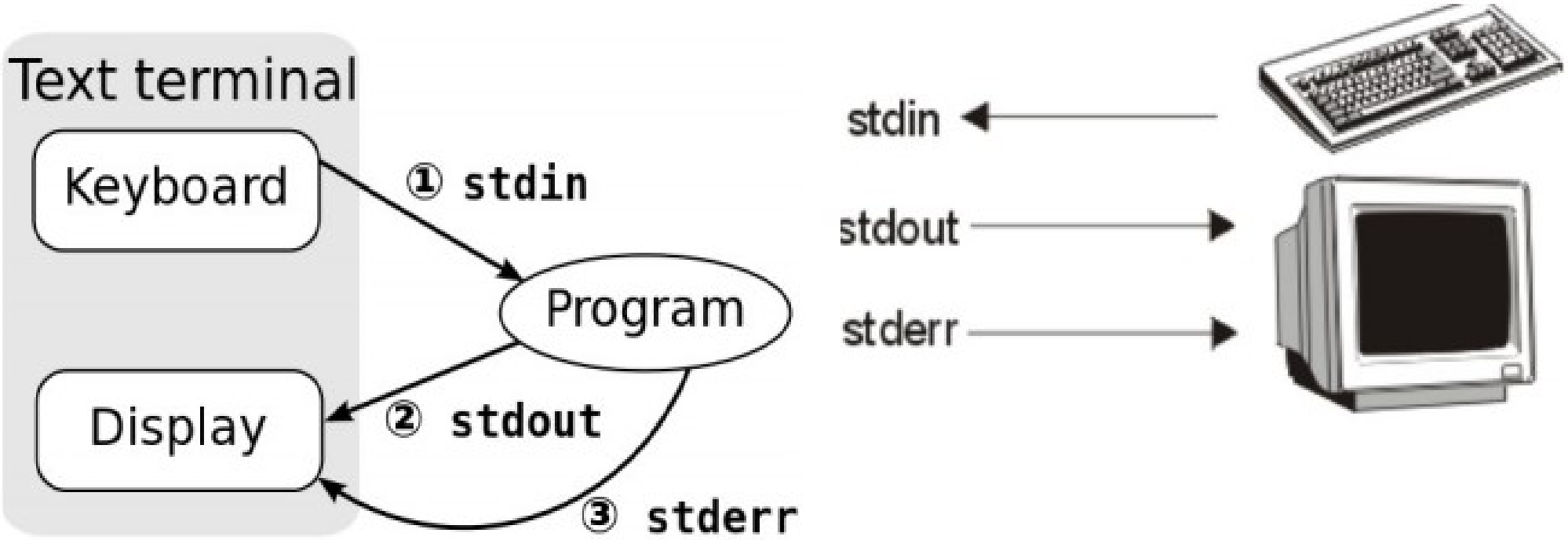
授课教师：李正华

2014.3.12

重定向

- 将程序的标准输入、输出、错误输出定向到其他位置

标准输入、标准输出及标准错误



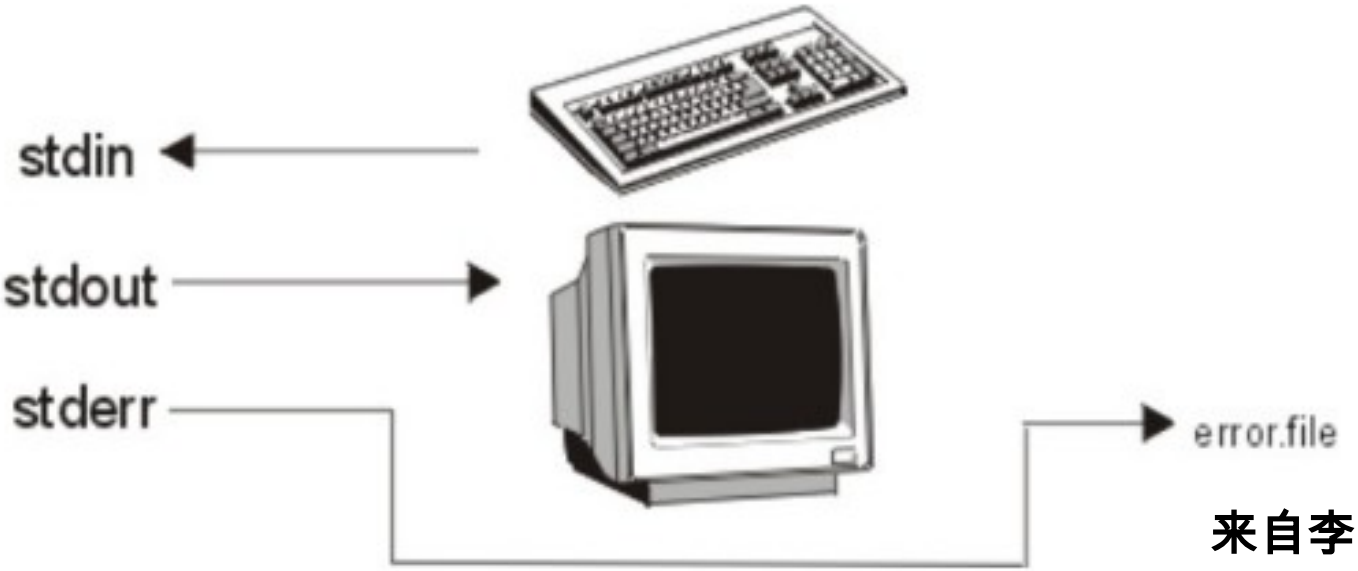
Linux/Unix 中任何进程都自动具备三个文件的操作权限：标准输入（stdin，读键盘），标准输出（stdout，写屏幕），标准错误（stderr，写屏幕）

来自李中国老师课件

输入输出重定向(redirection)

- 命令解释器 Shell 执行程序时，可以将程序的标准输入（或标准输出/标准错误）从键盘（或显示器）重定向至用户指定的其它位置（如文件）

command 2> error.file 的重定向示意图：



来自李中国老师课件

输出重定向

- ls -l > x （标准输出写到 x 中，覆盖）
- ls -l >> x（标准输出追加到文件 x 中）
- ls -l /root 2> x （标准错误输出写到 x 中
- find / -name *bin* > x 2> xx (分别输出）
- sort（ctrl + D 表示输入结束）
- sort < x
- cat
-

输入重定向

- **很多程序如 sort cat 如果不提供参数，那么默认从标准输入读入数据。这使得输入重定向稍微难理解一些。**
- sort（ctrl + D 表示输入结束）
- sort < x
- sort x

- cat 也是类似的， try yourself

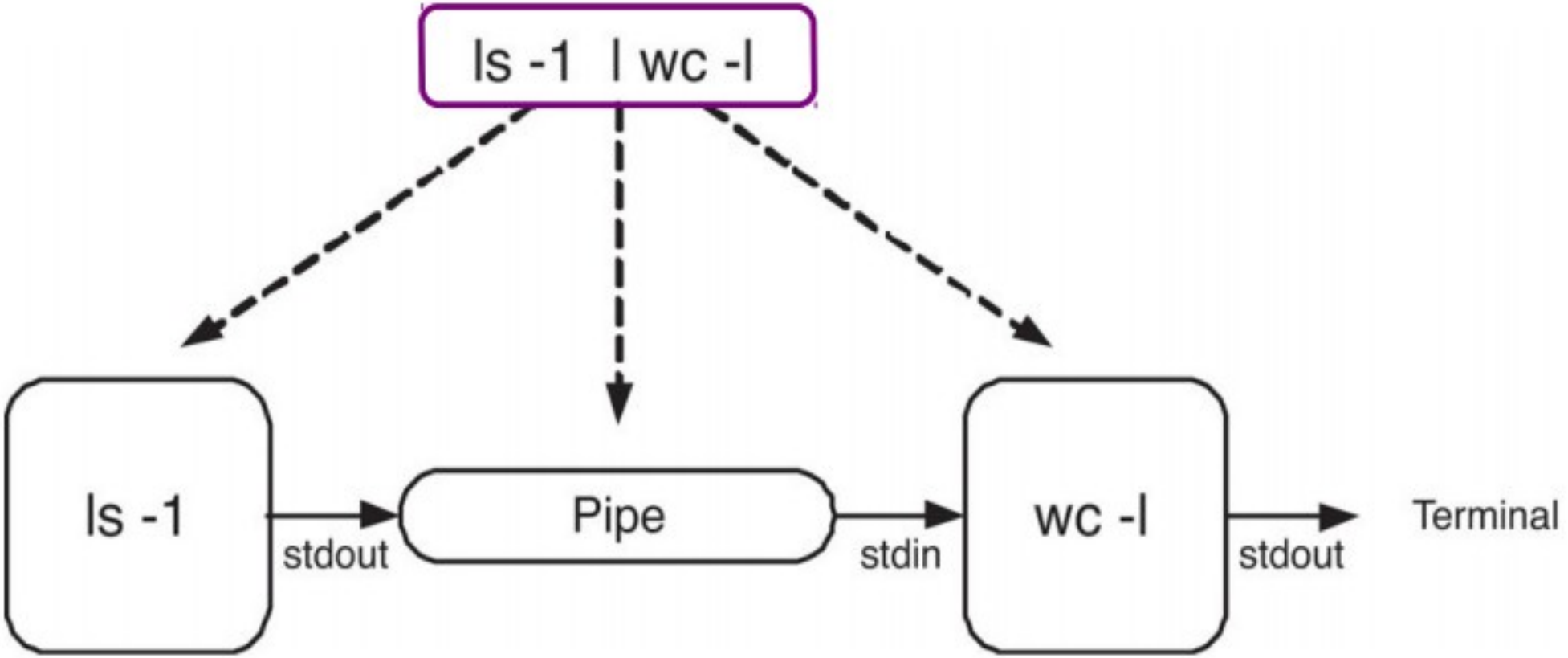
管道（ pipeline ）

- 打印 1 到 1000 以内所有素数？

管道（ pipeline ）

- seq 1 1000 | xargs factor | awk 'NF==2 {print $2}' | fmt

管道(pipeline)



管道机制：She11 可以将一个程序的标准输出重定向为另一个程序的标准输入，从而实现对数据流的多步处理和加工

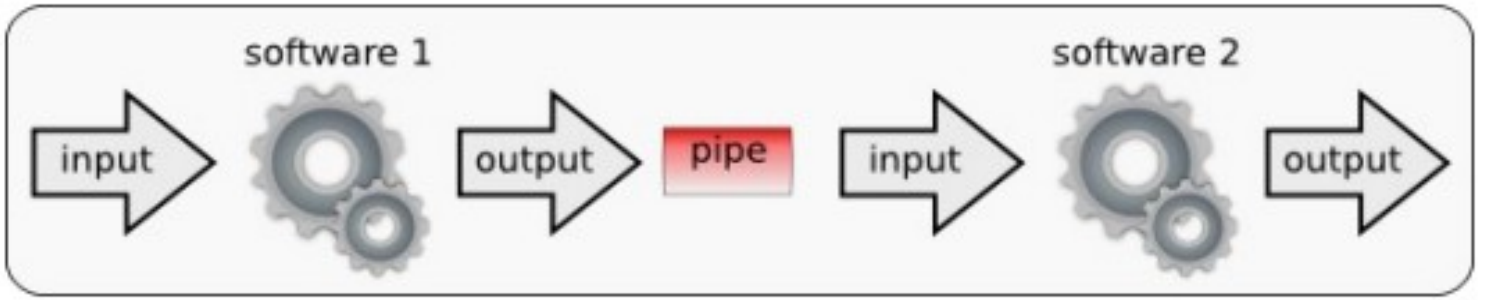
来自李中国老师课件

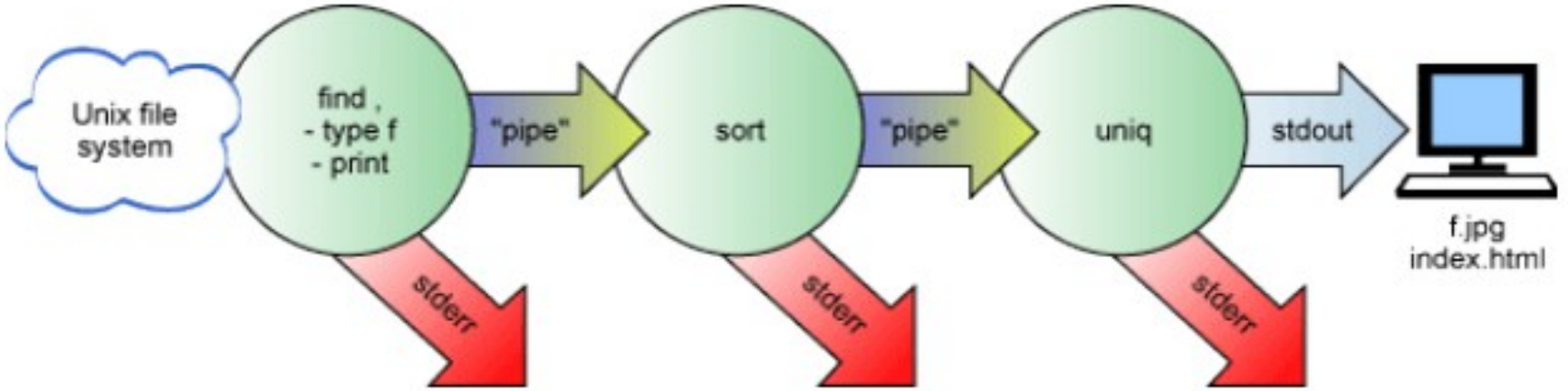
管道(pipeline)

- 管道是操作系统提供的进程间通信机制(IPC)
- 仅可用于进程间的单向通信

管道(pipeline)

- 需注意：管道只会把标准输出转为下一程序的标准输入；标准错误不会成为其它程序的标准输入





文件夹操作

- cd
- pwd
- ls
- mkdir hello-world
- rm hello-world -rf
- mv hello-world hello-world2

管道（ pipeline ）

- seq 1 1000 | xargs factor | awk 'NF==2 {print $2}' | fmt

- sort words.txt | uniq -c | wc -l
- cat words.txt | tr [:lower:] [:upper:] | sort | uniq -c | wc -l

Linux 下写 C 语言程序

- 程序 1 ，写到 a.c
  - 功能：输出 hello world
- 如何编译？

Linux 下写 C 语言程序

- 程序 1 ，写到 a.c
  - 功能：输出 hello world
- 如何编译？
  - gcc -c a.c
  - gcc a.o -o a
- 如何运行？

Linux 下写 C 语言程序

- 程序 1 ，写到 a.c
  - 功能：输出 hello world
- 如何编译？
  - gcc -c a.c
  - gcc a.o -o a
- 如何运行？
  - ./a
  - 环境变量： echo $PATH

Linux 下写 C 语言程序

- 程序 2
  - 功能：输出进程 id
  - <stdio.h> <unistd.h>
  - printf("", ...)
  - int getpid()

Linux 下写 C 语言程序

- 程序 3
  - 功能：从标准输入读入字符，小写变大写，输出到屏幕上
  - <stdio.h> <ctype.h>
  - char getchar()　[EOF 判断是否输入结束]
  - putchar(char)
  - char toupper(char)

Linux 下写 C 语言程序

- 程序 4
  - 功能：从**标准输入或文件**中读入字符，小写变大写，输出到屏幕上
  - 如果命令行参数数目 argc 为 2 ，则从第 2 各参数指定的文件中读入；否则从 stdin 读入。第一个参数 argv[0] 总是程序的名字。
  - int main(int argc, char *argv[])
  - FILE *fopen(char *)
  - fclose(FILE *)
  - char getc(FILE *)　[EOF 判断是否输入结束]

Linux 下写 C 语言程序

- 程序 5
  - 功能：从**标准输入或文件**中读入字符，小写变大写，输出到屏幕上
  - 命令行参数数目 argc 可以大于 2 ，则允许从多个文件读入

Linux 下写 C 语言程序

- 程序 6
  - 实现以下 cp 命令的工作，只局限于文件的拷贝。
  - 如 cp a.txt b.txt