

Pipes 2

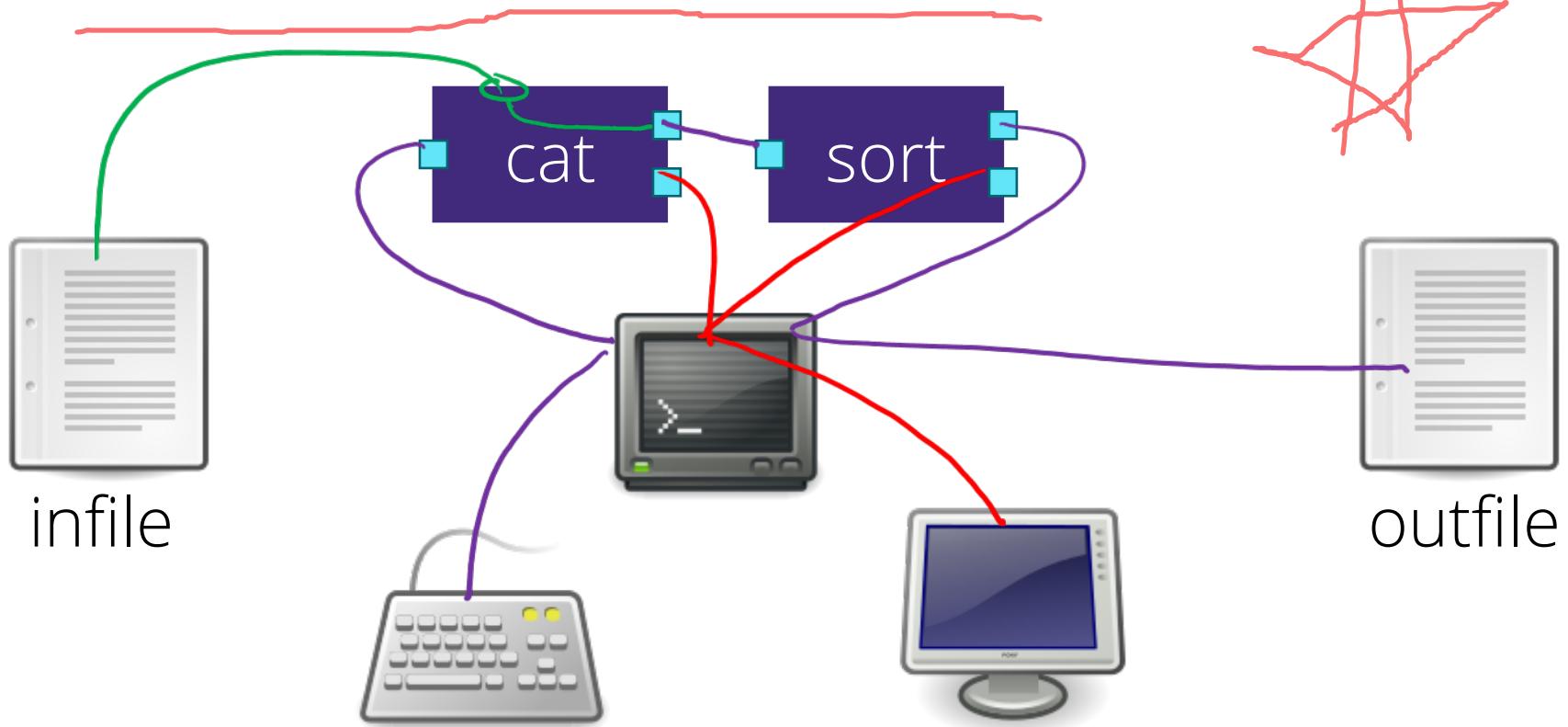
COMS10012 / COMSM0085

Software Tools

redirects

redirect

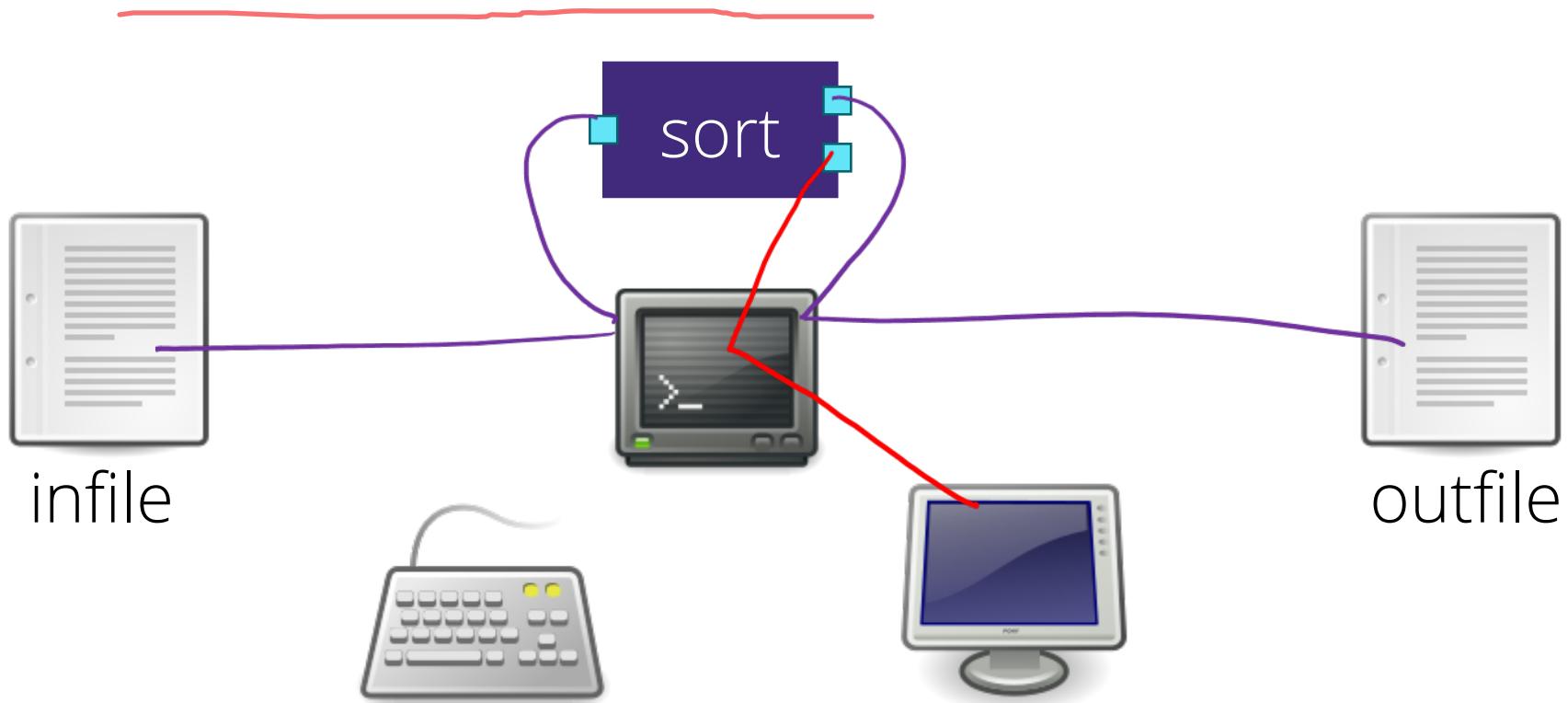
```
$ cat infile | sort > outfile
```



redirect

```
$ sort < infile > outfile
```

把 infile 的内容先排序好，然后输出到 outfile



redirect

\$ **COMMAND > FILE**

FILE

overwrites

\$ **COMMAND >> FILE**

FILE

appends to

error redirect

first command, the standard output of "COMMAND" will be saved in "FILE," and the standard error will be saved in "FILE2."

\$ COMMAND > FILE 2> FILE2

\$ COMMAND > FILE 2>&1

second command, both stdout and stderr from "COMMAND" will be combined and written to "FILE."



not:

\$ COMMAND 2>&1 > FILE



ignore output:

\$ COMMAND > /dev/null



files vs streams

A program that uses a standard stream can be told to use a file instead by

- PROGRAM < FILE (standard input)
- PROGRAM > FILE (standard output)
- PROGRAM 2> FILE (standard error)

files vs streams

A program that expects a filename can be told to use standard input/output instead by:

- using the filename - (single dash),
if the program supports it
- using the filename /dev/stdin etc.,
if your OS supports it

Filenames with dashes

Filenames starting with dashes are generally considered bad.

If you really want to address one (e.g. you created one by mistake), use e.g.

```
$ cat ./ -
```

```
$ rm ./ -f
```

advanced

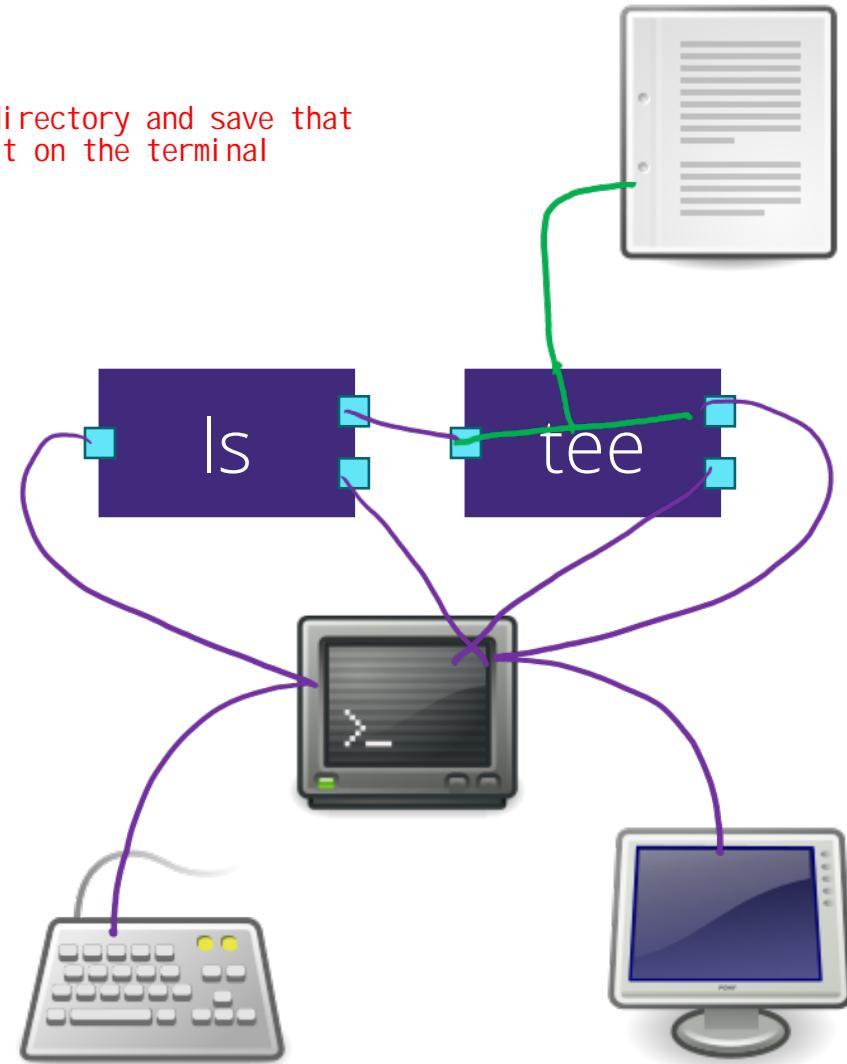
tee

```
ls | tee FILE:
```

it will list the contents of the current directory and save that list to the "FILE" while also displaying it on the terminal

```
$ ls | tee FILE
```

tee: takes a filename as argument and writes a *copy* of input to it, as well as to stdout



pgers

\$ ls | less

less is a pager: it displays text on your screen, one page at a time.

- Up/Down arrows scroll,
- Space/Enter advance a page,
- / (forward slash) opens a search,
- q quits.

This takes direct control of the screen (terminal).

sed

千万不能省略Uni verse 后边的 /

```
$ echo "Hello World" | sed -e  
's/World/Universe/'
```

Hello Universe

using the echo command to output the text "Hello World" to the standard output. It then uses the sed command to replace the word "World" with "Universe" in the input text

sed stands for stream editor – it can change text using a regular expression as it passes from input to output.

s/ONE/TWO/ [g] replaces the first match for ONE (all matches, with /g) with TWO. Regular expressions are supported.

need a file, want a pipe

If PROGRAM wants a file to read from, how can I pipe something in?

\$ **PROGRAM <(SOMETHING)**

\$ cat <(echo "Hi")
这里边 < (之间不能够有空格

Hi

\$ echo <(echo "Hi")
/dev/fd/63

echo "Hi" is a command that prints the text "Hi" to the standard output.

<(echo "Hi") is process substitution, where the output of echo "Hi" is treated as a temporary file-like object.

subshell

```
$ cat <(echo "Hi")
```



/dev/fd/...



subshell to argument

\$ COMMAND \$(SOMETHING)

\$ echo \$(echo Hi | sed -e s/Hi>Hello/)

Hello

echo \$(echo 'HI' | sed -e 's/HI>Hello/')

这条命令和上边的效果一模一样

old-fashioned way, with backticks:

\$ COMMAND ` SOMETHING `