

Sistemas Operativos

Redirecionamento de Descritores de Ficheiros

Grupo de Sistemas Distribuídos
Universidade do Minho

1 Objetivos

Familiarizar-se e utilizar as chamadas ao sistema relativas à duplicação de descritores de ficheiros e a sua aplicação no redirecionamento, mais especificamente dos descritores *standard* de entrada, saída e de erros.

2 Chamadas ao sistema

```
#include <unistd.h>          /* chamadas ao sistema: defs e decls essenciais */  
  
int dup(int fd);  
int dup2(int fd1, int fd2);
```

3 Exercícios propostos

1. Escreva um programa que redirecione o descritor associado ao seu *standard input* para o ficheiro `/etc/passwd`, e o *standard output* e *error* respetivamente para `saida.txt` e `erros.txt`. Imediatamente antes de o programa terminar, o mesmo deve imprimir no terminal a mensagem "terminei".
2. Modifique o programa anterior de modo a que, depois de realizar os redirecionamentos, seja criado um novo processo que realize operações de leitura e escrita. Observe o conteúdo dos ficheiros. Repare que o processo filho "nasce" com as mesmas associações de descritores de ficheiros do processo pai.
3. Modifique novamente o programa inicial de modo a que seja executado o comando `wc`, sem argumentos, depois do redirecionamento dos descritores de entrada e saída. Note que, mais uma vez, as associações – e redirecionamentos – de descritores de ficheiros são preservados pela primitiva `exec()`.
4. Escreva um programa que execute o comando `wc` num processo filho. O processo pai deve enviar ao filho através de um pipe anónimo uma sequência de linhas de texto introduzidas pelo utilizador no seu *standard input*. Recorra à técnica de redirecionamento de modo a associar o *standard input* do processo filho ao descritor de leitura do pipe anónimo criado pelo pai. Recorde a necessidade de fechar o(s) descritor(es) de escrita no pipe de modo a verificar-se a situação de *end of file*.
5. Escreva um programa que emule o funcionamento do interpretador de comandos na execução encadeada de `ls /etc | wc -l`.
6. Escreva um programa que emule o funcionamento do interpretador de comandos na execução encadeada de `grep -v ^# /etc/passwd | cut -f7 -d: | uniq | wc -l`.

4 Exercícios Adicionais

1. Acrescente ao interpretador de comandos proposto no guião 3 a possibilidade de redirecionar as entradas, saídas e erros dos comandos por ele executados. Considere os operadores de redirecionamento `<`, `>`, `>>`, `2>` e `2>>`.
2. Acrescente ao interpretador de comandos proposto nos guiões anteriores a possibilidade de encadear as entradas e saídas de programas a executar através de pipes anónimos (operador `|`).