

# Sistemas Operativos

## Gestão de Processos

Grupo de Sistemas Distribuídos  
Universidade do Minho

### 1 Objectivos


Familiarizar-se e utilizar as chamadas ao sistema relativas a criação e gestão de processos.

### 2 Chamadas ao sistema

```
#include <unistd.h>      /* chamadas ao sistema: defs e decls essenciais */
#include <sys/wait.h>     /* chamadas wait*() e macros relacionadas */

pid_t getpid(void);
pid_t getppid(void);
pid_t fork(void);
void _exit(int status);
pid_t wait(int *status);
pid_t waitPID(pid_t pid, int *status, int options);
int WIFEXITED(int status); /* macro */
int WEXITSTATUS(int status); /* macro */
```

### 3 Exercícios propostos

1. Implemente um programa que imprima o seu identificador de processo e o do seu pai. Comprove – invocando o comando `ps` – que o pai do seu processo é o interpretador de comandos que utilizou para o executar. 
2. Implemente um programa que crie um processo filho. Pai e filho devem imprimir o seu identificador de processo e o do seu pai. O pai deve ainda imprimir o PID do seu filho.
3. Implemente um programa que crie dez processos filhos que deverão executar sequencialmente. Para este efeito, os filhos podem imprimir o seu PID e o do seu pai, e finalmente, terminarem a sua execução com um valor de saída igual ao seu número de ordem (e.g.: primeiro filho criado termina com o valor 1). O pai deverá imprimir o código de saída de cada um dos seus filhos.
4. Implemente um programa que crie dez processos filhos que deverão executar em concorrência. O pai deverá esperar pelo fim da execução de todos os seus filhos, imprimindo os respectivos códigos de saída.

5. Pretende-se determinar a existência de um determinado número inteiro nas linhas de numa matriz de números inteiros, em que o número de colunas é muito maior do que o número de linhas. Implemente, utilizando processos um programa que determine a existência de um determinado número, recebido como argumento, numa matriz gerada aleatoriamente.
6. A partir do cenário descrito no exercício anterior, pretende-se que imprima por ordem crescente os números de linha onde existem ocorrências do número.

## **4 Exercícios Adicionais**

1. Implemente uma nova versão do programa feito nos exercícios anteriores que opere sobre uma matriz persistida em ficheiro (em formato binário). A matriz deve ser gerada aleatoriamente pelo processo pai e escrita em ficheiro, no início da execução do programa. Após este passo deve ser possível pesquisar a existência de um determinado número utilizando múltiplos processos.