



## Trabalho Nº 3 - Chamadas ao Sistema

### 1. Introdução

O Sistema Operativo é constituído por um conjunto de programas que implementam uma série de funcionalidades que constituem a interface entre as aplicações do Utilizador e o Hardware de um Computador.

Para invocar as funcionalidades do Sistema Operativo, as aplicações podem utilizar directamente uma interface aplicacional (API) do Sistema Operativo, designada por System Calls ou chamadas ao sistema.

Todos os programas que se executam num sistema acabam forçosamente por utilizar essa API quer directamente, quer através de bibliotecas de utilitários que introduzem funcionalidade adicional e de certo modo escondem alguma da complexidade associada aos System Calls.

Para se entender o funcionamento de um sistema operativo, uma das primeiras abordagens possíveis é interagir com ele sem intermediários.

Com este trabalho pretende-se que os alunos adquiram esse conhecimento, através da realização de programas que irão utilizar directamente alguns dos system calls que fazem parte do API do sistema Linux.

### 2. Trabalho a realizar

#### 2.1 Invocação directa de um System Call.

Utilizando a linguagem Assembler e a convenção de invocação dos system calls do Linux, escreva 3 programas que respectivamente:

- a<nº aluno>\_hello\_1.c: Escreve *Hello, World* no terminal
- a<nº aluno>\_hello\_2.c: Recebe uma string como parâmetro e imprime-a no terminal.
- a<nº aluno>\_hello\_3.c: Recebe uma string como parâmetro e escreve *Hello, <string recebida>*

Para invocar o assembler:

```
$ as hello.s -o hello.o
```

Para invocar o loader e criar o executável:

```
$ ld hello.o -o hello
```

#### 2.2 Utilização dos system calls open, read e write

Utilizando a interface aos system calls fornecida pela libc, escreva um programa em C que efectue a cópia de ficheiros, segundo a sintaxe:

```
$ copy source_file dest_file
```

- a) a<nº aluno>\_copy\_1.c: O programa deverá testar a existência do ficheiro de destino e imprimir uma mensagem de erro (“O ficheiro <nome do ficheiro> já existe.”) caso este

# Licenciatura em Eng.<sup>a</sup> Informática

## Sistemas Operativos - 2º Ano - 1º Semestre

já exista.

- b) a<nº aluno>\_copy\_2.c: O programa deverá aceitar uma flag (-overwrite), que especifica que, caso o ficheiro exista, deve ser substituído pelos novos dados.

Para conhecer a sintaxe de invocação das funções de biblioteca, pode utilizar as páginas do manual on-line do Linux através do comando:

```
$ man 2 <função>
```

Ou procurar no manual on-line no site:

<http://www.sourcentral.org/man/SUSE101>

### 3. Modelo de Entrega do Trabalho

O trabalho deverá ser realizado individualmente, e entregue até ao final do dia 11 de Novembro (Domingo). Os trabalhos poderão ainda ser entregues excepcionalmente até às 19h do dia seguinte (segunda-feira), sendo penalizados em 2 (dois) valores. Não serão aceites trabalhos entregues após essa data.

O trabalho deve ser enviado para o email [so2007.lusofona@gmail.com](mailto:so2007.lusofona@gmail.com) no seguinte formato:

Subject: a<nº aluno>-<dia aula>-<hora-aula>-trab3

(exemplo: o aluno 20171234, com aula na terça-feira às 14h, enviará o subject a2017134-3-14-trab3)

Anexos: a<nº aluno>-trab3.pdf, contendo um relatório esclarecedor do trabalho (limite de 3 páginas, fonte mínima 10).

a<nº aluno>-trab3.zip, contendo o código fonte das diversas alíneas (a<nº aluno>\_hello\_1.c, a<nº aluno>\_hello\_2.c, ...)

Critérios de avaliação, por ordem crescente de importância:

Funcionalidade

Organização e legibilidade do código

Relatório

Inovação não exigida no enunciado (1 valor)

### 4. Links e Referências

*The GNU Assembler*

<http://tigcc.ticalc.org/doc/gnuasm.html>

*Linux Assembly*

<http://asm.sourceforge.net>

*Linux System Calls Table*

[http://docs.cs.up.ac.za/programming/asm/derick\\_tut/syscalls.html](http://docs.cs.up.ac.za/programming/asm/derick_tut/syscalls.html)

“*The C Programming Language, 2nd Edition (Ansi C)*” de by Brian W. Kernighan, Dennis M. Ritchie, ed. Prentice Hall. <http://cm.bell-labs.com/cm/cs/cbook>

“*The C Book*, second edition, by Mike Banahan, Declan Brady and Mark Doran, ed. Addison Wesley [http://publications.gbdirect.co.uk/c\\_book](http://publications.gbdirect.co.uk/c_book)

*Manual de Referência (man pages) Linux on-line:* [www.sourcentral.org/man/SUSE101](http://www.sourcentral.org/man/SUSE101)