

**MEFT - Programação**  
**1º Ano - 1º Semestre de 2015/2016**  
**Série 8 (30/11/2015)**

1. Construa um programa que calcula numericamente a solução da equação diferencial:

$$\frac{dx}{dt} = -\lambda x$$

o utilizador deve fornecer, na linha de comandos, o valor de  $\lambda$ , o valor inicial de  $x$  ( $x_o$ ) e os valores iniciais e finais de  $t$ .

Com vista à representação gráfica dessa solução particular, escreva, num ficheiro, os resultados para que possam ser lido pelo 'Gnuplot', pelo 'Libre-Office', pelo 'OpenOffice' ou pelo 'Excel'.

2. Considere o problema 3 da série 6. Refaça-o utilizando um ficheiro binário. Deverá obter um ficheiro com cerca de 130 kB. Justifique o seu valor.

**Nota:** Para escrever, ler ou posicionar num ficheiro em binário, tem ao dispor as seguintes funções (Para mais detalhes sobre as funções ver manuais da biblioteca de C):

- `size_t fread` (`void *data`, `size_t size`, `size_t count`, `FILE *stream`)
- `size_t fwrite` (`void *data`, `size_t size`, `size_t count`, `FILE *stream`)
- `long int ftell` (`FILE *stream`)
- `int fseek` (`FILE *stream`, `long int offset`, `int whence`)

Em que “whence” é a posição a partir da qual o *offset* é calculado:

1. **SEEK\_SET**: a partir do início do ficheiro;
2. **SEEK\_CUR**: a partir da posição actual no ficheiro;
3. **SEEK\_END**: a partir do fim do ficheiro;

**3.** Faça uma implementação elementar da função 'ls' de unix (ou 'dir' do MS/DOS ou janela de comandos do Microsoft Windows), que mostra os ficheiros contidos numa directoria:

```
ls [-l] [ directoria ]
```

em que os [...] indicam que se pode omitir essa informação, "*directoria*" representa a pasta de que se pretende listar o conteúdo e "-l" significa que se deverá acrescentar a essa informação o número de bytes do ficheiro e a data em que foi alterado pela última vez.

O resultado final deverá ser uma lista em que cada linha apresenta um ficheiro diferente e, no caso da opção "-l", o seu aspecto deve ser o seguinte:

```
273635 Nov 16 09:34 ficheiro.ext
```

Para tal deverá usar as funções:

- **opendir** ('dirent.h'): Abre uma directoria para leitura. Vagamente equivalente a 'fopen' para ficheiros.
- **readdir** ('dirent.h'): cada vez que é chamada retorna um ponteiro (struct dirent \*) para a informação do membro da directoria em causa.
- **stat** (para windows) e **lstat** (para unix) ('sys/stat.h'): dão a informação referente a um dado ficheiro.
- **closedir** ('dirent.h'): fecha uma directoria aberta com 'opendir'.
- **localtime** ('time.h'): recebe como argumento um ponteiro para o tempo 'time\_t' e retorna um ponteiro para a estrutura 'struct tm'.
- **strftime** ('time.h'): idêntica a *sprintf* para escrever o tempo numa string.

**Nota1:** Para mais detalhes sobre as funções ver manuais da biblioteca de C.

**Nota2:** o programa deverá poder ser compilado em unix ou em windows fornecendo os resultados correctos.