



## Programação 2

2º Semestre

### Trabalho Prático P1A

Efectue as tarefas de programação descritas abaixo, usando a linguagem C++ em ambiente Linux.

Pretende-se escrever um programa em C++, denominado “ops”, que executa operações matemáticas. O código fonte do programa deve ser escrito no ficheiro com o nome “ops.cpp”, e deve ser compilado com o comando:

```
g++ ops.cpp -o ops
```

Construa o programa de acordo com as alíneas que se seguem. Teste sempre o programa no final de cada alínea.

- a) [3.5 valores] O programa deve ler do *standard input* dois números (separados por espaço) e escrever no ecrã o resultado do primeiro número levantado à potência do segundo. Por exemplo, para os números 3 e 2, deve dar o resultado 9.

Sugestão: usar a função `double pow(double base, double exp)` já definida na biblioteca matemática “math.h”.

Experimentar o programa com entrada de dados do teclado. Para sinalizar o fim da entrada de dados a partir do teclado, deve digitar `ctrl-D`.

- b) [2.5 valores] O programa deve ser invocado com um argumento, o qual indica qual a operação a efectuar – devem ser suportadas as operações `mul` (multiplicação), `div` (divisão inteira), `soma` (soma), `sub` (diferença) e `pot` (potência). Se a chamada for realizada sem argumentos, o programa deve imprimir uma mensagem de erro. Para aceder aos argumentos passados na linha de comando, é necessário declarar

```
int main(int argc, char *argv[])
```

em que `argc` é o número de argumentos (incluindo o nome do programa) e `argv` é o array de argumentos. Assim, ao chamar `ops mul`, `argc` é 2, `argv[0]` é “ops” e `argv[1]` é “mul”

- c) [1.5 valores] O programa deve aceitar um argumento extra, o qual é o nome do ficheiro onde se encontram os números a operar. Exemplo de ficheiro:

```
3 5
2 3
```

- d) [2.5 valores] Declare um vector de inteiros, e, para cada operação, insira o resultado no final do vector (sugestão: utilize a função `push_back()`). No final de todas as operações, imprima os resultados por ordem (isto é, todos os valores do vector).