

Exercício 00: Introdução ao Python, ao Mathematica, e ao LaTeX

Não tem que ser feito relatório. Este exercício está orientado para C++, mas tudo pode ser feito usando outra linguagem de programação. Pode usar editores texto e para gráficos pode usar softwares externos como gnuplot ou xmgrace.

0.1. Introdução ao Linux (WSL para utilizadores de Windows)

Um terminal em Linux é um interface de texto que permite escrever e executar comandos. O objetivo desta tarefa é explorar alguns dos comandos mais simples.

1. Abra um terminal (shell);
2. Crie uma nova pasta *teste*: `mkdir teste`
3. Entre na diretoria: `cd teste`
4. Inicie um software de edição de texto para escrever o programa em C++ (por exemplo, leafpad, vim ou emacs): `leafpad test.txt`
5. Escreva qualquer coisa, grave o ficheiro e saia do programa.

Outros comandos úteis de Linux são:

Comando	Utilidade
<code>pwd</code>	mostrar o caminho para a diretoria atual
<code>mkdir name</code>	criar a diretoria "name"
<code>cd name</code>	entrar na diretoria "name"
<code>gedit</code>	entrar no editor de texto
<code>ls</code>	listar os ficheiros e diretorias
<code>ls -l</code>	incluir mais informação na listagem
<code>cp, mv</code>	copiar e mover ficheiros/diretorias
<code>man</code>	abrir o manual de um comando
<code>g++</code>	compilar programas em C e C++
<code>command &</code>	executar um comando em "background"
<code>exit</code>	fechar o terminal

0.2. Introdução ao C++

O objetivo desta tarefa é escrever um primeiro código em C++.

1. Abra um terminal (shell);
2. Crie uma nova pasta *hello*: `mkdir hello`
3. Entre na diretoria: `cd hello`
4. Inicie um software de edição de texto para escrever o programa em C++ (por exemplo, leafpad, vim ou emacs): `leafpad hello.cpp`
5. Escreva o seguinte código:

```
#include <iostream>
using namespace std;
```

```
int main ( )
{
    cout << "Hello World." << endl;
    return 0;
}
```

6. Compile o código: `g++ hello.cpp -o hello.x`

7. Corra o código: ./hello.x
8. Modifique o texto de saída, compile e corra o programa de novo.

0.3. Array simples (C++)

Implemente um programa que defina um *array* com os seguintes valores (aproveite o código da aula anterior):

```
{10.5, 9.3, 11.4, 10.9, 13.0, 8.4, 9.2, 8.9, 10.3, 11.2, 12.1, 8.4, 9.2, 9.9, 10.1}
```

O programa deve correr o *array* e imprimir para o ecrã de dois em dois valores.

O tipo de *array* de números reais mais simples é criado com:

```
double lista[15]; // cria um array de tamanho 15 (lugares 0 a 14)
```

Para escrever numa posição do *array* deve-se usar:

```
lista[2]=11.4; // escreve o valor 11.4 na posição 2 do array lista
```

0.4. Cálculo de valores médios e desvio padrão (C++)

Modifique o programa anterior para calcular as seguintes quantidades:

1. Valor médio dos números da lista $\langle x \rangle = \frac{1}{N} \sum_{i=1}^N x_i$
2. Variância: $Var = \frac{1}{N} \sum_{i=1}^N (x_i - \langle x \rangle)^2$
3. Desvio padrão: $\sigma = \sqrt{Var}$

0.5. Escrita de dados em ficheiros (C++)

Imprimir para um ficheiro todos os valores no *array* do primeiro exercício. Como exemplo de escrita em ficheiros:

```
ofstream outfile; // cria um objeto chamado outfile
outfile.open("nome.txt"); // abre um ficheiro chamado nome.txt
outfile << 1 << "\t" << 10 << endl; // escreve para um ficheiro os números 1 e 10
outfile.close(); // fecha o ficheiro
```

Para usar estas funções é necessário incluir no início do código:

```
#include <fstream>
```

0.6. Leitura de dados em ficheiros (C++)

Fazer a leitura do ficheiro criado no exercício anterior e testar enviando para o ecrã.

```
ifstream infile;
infile.open("nome_do_ficheiro.txt");
infile >> var1; // lê o primeiro valor do ficheiro para a variável var1
infile >> var2; // lê o segundo valor do ficheiro para a variável var2
infile.close();
```

0.7. Mais alguns exemplos simples em C++

Programa para gerar uma tabela com os valores dados por uma parábola (parabola.cpp).

```
#include <iostream>
using namespace std;
```

```
int main ( )
```

```

{
  for (int i=1; i<=10; i++)
  {
    cout << i << "\t" << i*i << endl; //imprime o valor de i e do seu quadrado
  }
  return 0;
}

```

Corra o programa guardando a informação de saída para um ficheiro parabola.dat :
`./parabola.x > parabola.dat`

Faça o gráfico dos dados usando, por exemplo, o xmgrace ou o gnuplot. Um exemplo para fazer o gráfico do ficheiro e comparar com uma parábola é:

```
plot "parabola.dat" u 1:2 title 'Data', x**2 title 'Function'
```

Faça o mesmo para outras funções, por exemplo: $\sin(x)$, $\log(x)$ e $\exp(x)$. Para utilizar estas funções é necessário incluir no início do código:

```
#include<math.h>
```

0.8. Introdução ao Mathematica

O objetivo desta tarefa é mostrar que muitas problemas podem ser resolvidos numericamente com software fechado (sem precisar programação). Neste caso vamos usar o Mathematica. Devem fazer os seguintes passos:

1. Iniciar o Mathematica
2. Criar uma função x^2 , escrevendo:
 - a. `F[x_]=x*x`
 - b. Carregar SHIFT+ENTER
3. Fazer o gráfico da função entre 0 e 10, escrevendo:
 - a. `Plot[F[x],{x,0,10}]`
 - b. Carregar SHIFT+ENTER

Faça o mesmo para outras funções, por exemplo: $\sin(x)$, $\log(x)$ e $\exp(x)$.

0.9. Array simples (Mathematica)

Criar o mesmo array que no exercício 0.1 escrevendo:

```
Lista={10.5, 9.3, 11.4, 10.9, 13.0, 8.4, 9.2, 8.9, 10.3, 11.2, 12.1, 8.4, 9.2, 9.9, 10.1}
```

0.10. Cálculo de valores médios e desvio padrão (Mathematica)

Fazer a média, a variância e o desvio padrão usando os seguintes comandos:

1. `Mean[lista]`
2. `Variance[lista]`
3. `MeanDeviation[lista]`

Verifique se o resultado é o mesmo que me C++. Se não for igual, tente encontrar uma justificação.

0.11. Escrita de dados em ficheiros (Mathematica)

Para criar o mesmo ficheiro que no exercício 0.3 utilizar o comando:

```
Export["nome.txt",{1,10},"Table"]
```

Este comando cria um ficheiro nome.txt na diretoria predefinida (caso queira mudar a diretoria deve colocar o caminho todo). Com a opção "Table" cria uma lista de duas colunas. Caso queira mais linhas deve colocar $\{\{1,10\},\{2,20\},\{3,30\}\}$ (este comando cria 3 linhas com duas colunas).

0.12. Leitura de dados em ficheiros (Mathematica)

Para ler o ficheiro de volta para uma tabela deve usar o comando:

```
tabela = Import["nome.txt", "Table"]
```

Para ler cada valor da lista pode escrever:

```
tabela[[1, 1]]  
tabela[[1, 2]]
```

0.13. Introdução ao LaTeX

Fazer download do template de relatório que se encontra no Fenix. Para uns teste iniciais, pode usar o overleaf (<https://www.overleaf.com/>). Os PC da sala estão equipados com LaTeX. Para instalar em PC próprio pode usar o TeXworks (<http://www.tug.org/texworks/>) ou Kile (<https://kile.sourceforge.io/>). Para ter uma versão completa com os pacotes todos necessários, talvez instalar o MiKTeX (<https://miktex.org/download>).