

Paradigmas de Programação 1

Guia das Aulas Práticas

2003-2004

Miguel Oliveira e Silva

Iouliia Skliarova

Aula 1

Objectivos: Revisões de *C* e programação modular.
Introdução ao ambiente de programação do Visual Studio *C++*.

Problema A1

Crie um projecto (*Windows Console Application - Empty Project*) no Visual *C++* e implemente um programa em *C*, que leia um número inteiro e verifique se ele é primo.

Problema A2

Faça um programa que leia uma série de números inteiros positivos (a leitura termina quando se lê o número zero, ou quando se atinge os 10 números), e os ordene por ordem crescente.

Problema A3

Faça um programa que, de uma forma interactiva (recorrendo a um menu), permita gerir uma lista de pessoas. Cada pessoa é caracterizada pelo nome, número do BI e idade. Numa primeira versão o programa deve permitir a introdução de novas pessoas na lista, apagar pessoas existentes e apresentar a lista completa (com uma formatação visual minimamente aceitável).

A3.1. Acrescente a possibilidade de ordenar a lista por nome, ou por número de BI.

A3.2. Acrescente a data de nascimento à informação de cada pessoa.

Problema B1

Construa a classe `CShere` que represente uma esfera e contenha a informação sobre o raio da esfera (inicializado no construtor). Implemente métodos que permitam consultar e modificar o raio.

A estrutura da classe pode ser a seguinte:

```
class CSphere
{
public:
    // construtor
    // destrutor

    // métodos

private:
    double m_dRadius;
};
```

B1.1. Implemente métodos que permitam calcular o diâmetro, circunferência, área da superfície e volume da esfera.

B1.2. Construa um método que visualize no écran toda a informação sobre a esfera, i.e. raio, volume, etc.

B1.3. Faça um programa que exemplifique o uso da classe CSphere.

Problema B2

Construir um “array” estático com a ajuda da classe CArray que tem a seguinte estrutura:

```
class CArray
{
    // escrever um valor no array usando um índice
    // ler um valor do array usando um índice
    // imprimir todos os valores do array
    // devolver o tamanho do array
}
```

B2.1. Construir uma função global que tenha como único argumento um ponteiro para CArray.

B2.2. Usar o “array” passado no corpo da função, por exemplo, preencher cada posição com o valor do índice respectivo.