

Instituto Superior de Engenharia de Coimbra
CTSP TPSI/DW
Programação Orientada a Objectos
Exercícios

Ficha de exercícios 4

Tópicos Abrangidos	<ul style="list-style-type: none">• Objectos• Abstracção e representação de entidades através de classes.• Encapsulamento
-----------------------	---

Nota: Os exercícios desta disciplina e constantes nestas fichas destinam-se adquirir competência em programação orientada a objectos com C#. Os exercícios só funcionam se de facto os tentar fazer. Muito dificilmente conseguirá fazer a disciplina sem praticar. Não assuma que os docentes disponibilizarão soluções de forma sistemática. Deve tentar realizar os exercícios propostos e interagir com os docentes para esclarecer dúvidas e ouvir as soluções que forem propostas.

A falta de participação e atitude negativa nas aulas contará negativamente na nota.

1. Construa uma classe Pessoa cujos objectos representam pessoas da vida real. Neste exercício vai usar a característica de encapsulamento, impedindo o programa de interagir directamente com os dados da classe. A classe deve proporcionar mecanismos e não necessariamente dados. Exemplo: um objecto de pessoa comporta-se como tendo uma informação interna "idade", mas pode ser o caso de esse campo interno não existir de todo, havendo em vez disso um campo "data de nascimento".
 - a) Identifique os dados associados a pessoas relevantes para constarem na classe.
 - b) Identifique qual o estado interno (valores) aceitável que deve existir inicialmente nos objectos desta classe tendo em atenção que estes objectos representam pessoas da vida real. Defina os métodos adequados para garantir que não existirão objectos que não cumpram esse estado inicial.
 - c) Defina métodos adicionais para cumprir a funcionalidade típica de colocação e obtenção de dados.

2. Defina uma classe Automovel que contenha os dados necessários à representação de um automóvel (matrícula, combustível, marca, modelo, ou outros). No decorrer das alíneas seguintes deve usar uma função main que lhe permita testar o código que for fazendo. A função main não faz directamente parte do exercício.
- a) Analise o problema e determine i) que dados devem pertencer à classe e ii) quais os mecanismos e regras que devem ser materializados em funções para lidar com os objectos desta classe. As suas considerações devem incluir não só que dados e de que tipo, mas também que operações são permitidas (por exemplo, cilindrada menos que 50? Pode-se mudar a matrícula? Posso atestar o carro com ele em funcionamento? Deverá existir uma função mudaVelocidade() ou será melhor uma acelera() e outra trava()?). Debata as suas conclusões com o professor.
 - b) Escreva a classe de acordo com o que planeou na alínea anterior.
 - c) Escreva uma função para preencher o conteúdo de um objecto Automovel com valores fornecidos pelo utilizador. A função não é membro da classe. Nesta alínea evite criar objectos desnecessariamente. Nota: evite interagir com o utilizador directamente dentro das funções membro da classe.
 - d) Escreva uma função principal que permita usar e testar objectos da classe Automovel.
3. Construa uma classe EuroMilhoes, cujos objectos representam apostas de euromilhões. A classe deve representar fielmente o comportamento deste conceito da vida real, tanto a nível de dados como de mecanismos (ou seja, o que se pode fazer com os seus objectos e de que forma). Assim, os objectos de EuroMilhoes devem comportar-se como tendo informação interna acerca de números (5 marcações de números de 1 a 49 e 2 marcações de 1 a 9), e proporcionar mecanismos para:
- Garantir a correcta inicialização: i) sem nenhuma informação especificada, ii) dados um conjunto de 5 números no máximo.
 - Acrescentar (marcar) um novo número,
 - Remover (desmarcar) um número
 - Dados um conjunto de sete números (5 números e duas estrelas), verificar se a aposta corresponde ao primeiro prémio, segundo, terceiro (basta verificar estes).
 - Verificar se duas apostas são iguais.
- a) Elabore a classe descrita.
 - b) Elabore um programa que permita usar os objectos da classe: usando um pequeno menu, o utilizador deve poder criar, modificar e consultar uma aposta
 - c) Modifique o programa para poder trabalhar com até 10 apostas.

4. Construa uma classe ContaBancaria, cujos objectos representam contas bancárias. A classe deve representar o comportamento normal de uma conta bancária, tanto a nível de dados como de mecanismos. Assim, os objectos ContaBancaria devem comportar-se como tendo informação interna acerca da conta: Número, Nome do Cliente, Saldo e Estado, e proporcionar mecanismos para:

- Garantir a correcta inicialização:
 - Sem nenhuma informação especificada;
 - Fornecendo o n. da conta, o nome do cliente e o saldo inicial;
- Depositar um determinado montante;
- Levantar um determinado montante;
- Transferir um determinado montante para outra conta bancária;
- Consultar o saldo disponível;
- Consultar o número da conta;
- Consultar o nome do cliente;

Deve implementar regras que evitem comportamentos anómalos, como por exemplo, não deve ser permitido efectuar levantamentos se montante a levantar for superior ao saldo existente.

- a) Desenvolva um programa que teste o funcionamento da classe ContaBancaria. Para isso o programa deverá criar 4 contas diferentes, realizar diversas operações de depósito, levantamentos e transferências. No final o programa deverá apresentar um relatório com o n.º de conta, nome do cliente e saldo disponível para cada uma das contas.

5. Construa uma classe Banco, cujos objectos representam instituições bancárias. A classe deve representar, de forma resumida, o comportamento normal de um banco, tanto a nível de dados como de mecanismos. Assim, os objectos Banco devem comportar-se como tendo informação interna acerca do: Nome do banco, ContasBancarias, e proporcionar mecanismos para:

- Garantir a correcta inicialização:
 - o Sem nenhuma informação especificada;
 - o Fornecendo o nome do banco e o número máximo de contas possíveis de criar;
- Adicionar uma conta Bancária;
- Remover (desactivar) uma conta Bancária;
- Mostrar o número de contas bancárias existentes;

- a) Desenvolva um programa que teste o funcionamento da classe Banco. Para isso o programa deverá criar um Banco, realizar diversas operações de criação e remoção de contas bancárias. No final o programa deverá apresentar um relatório com descrição do banco, bem como um resumo das contas bancárias existentes.

6. Usando a classe EuroMilhoes do exercício anterior, construa uma classe que represente um boletim de apostas de euromilhões. Um boletim tem sempre 10 apostas e deve ter os mecanismos para inicialização, modificação e consulta das apostas individuais.

- a) Elabore a classe pretendida. Debata com o professor os méritos e desméritos de ter a funcionalidade de boletim vertida na função main, como estava no exercício anterior, ou encapsulado numa classe autónoma, como no caso deste exercício.