

## Trabalho Prático

As regras que regem o trabalho prático encontram-se descritas na ficha de unidade curricular, chamando-se a atenção para a sua leitura. O trabalho é composto por duas entregas, a Meta 1 e a Meta Final, sendo ambas sujeitas a apresentação/defesa presencial e individual. Existem pormenores relativos à apresentação e defesa do trabalho que irão ser atempadamente detalhados no decorrer do semestre.

### 1. Tema geral: Bolsa de valores online

O trabalho prático consiste na implementação de vários programas que, no seu conjunto, constituem uma bolsa de valores online. Naturalmente não se pretende algo com um elevado grau de realismo, mas apenas uma funcionalidade básica que permite explorar os assuntos e matéria desta disciplina.

No âmbito deste trabalho, todos os intervenientes serão **processos** a correr **na mesma máquina** e não é necessária comunicação em rede – apenas **comunicação entre processos**. Os programas envolvidos terão interfaces principalmente em consola, existindo a hipótese de interface gráfica como complemento. Todos os processos serão lançados pelo mesmo utilizador da conta do sistema operativo, não devendo confundir-se o utilizador do sistema da bolsa com o utilizador do sistema operativo. Para simular diferentes utilizadores da bolsa, o utilizador do sistema operativo lançará simplesmente vários processos, usando várias consolas.

### 2. Lógica geral e elementos do sistema de bolsa de valores

O que significa de forma centralizada?

A bolsa de valores tem um comportamento, apesar de simplificado, semelhante ao conceito comum de bolsa de valores. Na bolsa são transacionadas ações de empresas. As empresas, as ações disponíveis para compra/venda e **o preço de cada ação são controlados de forma centralizada**. As empresas, o número de ações inicialmente disponíveis e o preço inicial são especificados pelo **administrador da bolsa**. As operações de compra e venda são feitas a pedido dos **utilizadores da bolsa**, e cada utilizador está previamente registado na bolsa, sendo titular de uma carteira de ações e de um saldo, o qual é usado nas operações de compra e venda. As operações de compra e venda afetam o preço das ações numa lógica de lei de procura e oferta cujos detalhes são dados mais adiante. O preço pode também ser manipulado diretamente por ação do administrador. A bolsa permite operações de compra e venda, registo de novas empresas, listagem de informações diversas (empresas, utilizadores, etc.).

O sistema tem os seguintes elementos, aos quais correspondem programas:

- **Programa *bolsa***. Concretiza a agência central da bolsa que regula toda a gestão de dados e de operações. Apenas existirá uma execução deste programa num determinado instante. O programa bolsa interage com um utilizador “administrador da bolsa” que pode executar algumas operações de controlo indicadas mais adiante e que tomam a forma de comandos escritos (consola). O programa bolsa interage com outros programas, usando diversos mecanismos da forma especificada mais abaixo, tendo essa especificação carácter obrigatório.
- **Programa *cliente***. Este programa concretiza a interface de utilizador entre o sistema bolsa e o utilizador comum. Obtém ordens do seu utilizador na forma de comandos escritos (consola) e toda a interação com o utilizador é feita segundo o paradigma de consola de texto. O programa cliente interage com o bolsa por **named**

**pipes.** O programa cliente estará sempre apto a reagir a ordens indicadas pelo seu utilizador, e ao mesmo tempo, reagirá de imediato a qualquer comunicação que lhe seja dada pelo bolsa.

A relação entre o programa bolsa e o programa cliente é a de cliente-servidor, sendo o bolsa o servidor. **Num dado instante, por cada utilizador da bolsa haverá um programa cliente em execução.** A utilização deste programa pressupõe um momento inicial de autenticação em que o cliente envia as credenciais do seu utilizador (*username* e *password*) ao bolsa, estando já esse utilizador previamente registado na bolsa. A execução mantém-se até que o utilizador decida encerrar a sua execução, ou seja, durante a mesma execução são efetuadas várias operações. Apenas podem estar ligados ao bolsa, em simultâneo, NCLIENTES programas cliente. Caso se pretendam executar mais do que NCLIENTES clientes, os restantes devem manter-se em execução e aguardar por uma vaga. O valor NCLIENTES encontra-se especificado no *Registry*. Na primeira execução do bolsa, caso este valor não esteja definido, **deve ser utilizado o valor 5.**

- **Programa board.** Este programa apresenta uma lista das N empresas cujas ações são mais valiosas, apresentadas por ordem decrescente de valor. Apresenta também os dados da última transação realizada: nome da empresa, número de ações e valor. Todas as informações apresentadas são atualizadas (automaticamente). O valor de N é especificado através de um argumento da linha de comandos, estando limitado ao máximo de 10. Podem existir várias execuções em simultâneo deste programa e não é necessária nenhuma autenticação prévia. A comunicação entre o **board** e o **bolsa** é obrigatoriamente por **memória partilhada**.
- **Programa boardGUI.** Este programa tem uma natureza semelhante ao **board** na medida em que apenas apresenta informação. A sua execução também não está associada a um utilizador específico e podem existir múltiplas execuções em simultâneo. No entanto, a interface é gráfica e a informação é apresentada como um gráfico de barras (mais pormenores na Secção 4). A comunicação entre o **boardGUI** e o **bolsa** é feita obrigatoriamente por **memória partilhada**. A construção deste programa é **opcional**, podendo a sua elaboração compensar outros aspetos menos bem feitos ou omissos nos outros programas. Os pormenores desta natureza opcional são dados mais adiante.

### 3. Lógica de funcionamento – pormenores adicionais

- **Utilizadores.** Os utilizadores têm contas, que inicialmente são lidas de um ficheiro de texto (pelo bolsa). Não será preciso fazer nenhuma funcionalidade de adicionar/remover utilizadores. O nome do ficheiro que tem os utilizadores é dado pela linha de comandos na execução do bolsa. O formato do ficheiro é o seguinte: um utilizador por linha com a seguinte configuração: *username password saldo*. O *username* e a *password* são uma palavra apenas cada um. Exemplo:

```
maria qwerty 105.4
ze 123abc 66
```

**Pode ser assumido que nunca irão existir mais do que 20 utilizadores registados nem haverá usernames repetidos.**

- **Oferta inicial de empresas.** As empresas, o número de ações iniciais e o preço inicial de cada ação são indicados pelo administrador da bolsa através de comandos que este escreve na consola do programa bolsa. As ações inicialmente existentes ficam imediatamente disponíveis para venda após a abertura da bolsa. Adicionalmente, a criação de empresas e ações pode ser lida de um ficheiro de texto, através de um comando,

o que simplificará o uso e teste da aplicação. O ficheiro terá uma organização semelhante ao ficheiro de utilizadores: uma empresa por linha com a seguinte configuração: nome-empresa número-ações preço-ação. O nome da empresa tem apenas uma palavra e **não haverá nomes de empresas repetidos**. Exemplo:

metromondego 1000 0.01
spaceY 200 500

A leitura deste ficheiro acrescenta a informação à que já existir em memória (por exemplo, empresas criadas por comandos). Pode assumir que nunca haverá necessidade de guardar mais do que 30 empresas ao todo.

- **Carteira de ações.** Cada utilizador tem um conjunto de ações de empresas, ou seja, uma “carteira de ações”, que tem informações de número de ações de cada empresa onde esse utilizador participa. Um utilizador não poderá ter ações em mais do que 5 empresas diferentes. A estrutura de dados que representa esta informação pode ter mais dados que forem considerados úteis. Inicialmente um utilizador não tem ações em empresa nenhuma (o ficheiro de utilizador nem sequer tem essa informação).
- **Operações de compra.** As operações de compra são pedidas pelos utilizadores através do programa cliente, que comunica com o bolsa, que é quem efetivamente faz essas operações. As operações de compra implicam um saldo suficiente desse utilizador quando a compra é concretizada. As compra pode ser satisfeita por ações que existam (sobrem) da oferta inicial quando a empresa foi registada, ou por oferta de venda que haja pendente nesse instante. Se o saldo do comprador for insuficiente, a compra é abortada para a totalidade das ações que estivessem a ser transacionadas e a operação não fica pendente.
- **Operação de venda.** Tem um comportamento semelhante à operação de compra, revertendo o valor da transação para o saldo do(s) vendedor(es).
- **Variação de preços de ações.** A compras fazem aumentar o preço e as vendas fazem baixar (após a venda/compra), segundo uma lógica natural de procura/oferta. Os pormenores são deixados ao critério dos alunos e a única coisa que há a dizer é que é obrigatório que haja essa variação. A variação pode também ocorrer de forma aleatória, simulando variações que ocorrem periodicamente, tal como no mundo real. Mais uma vez, os detalhes são deixados ao aluno. O preço pode também ser modificado administrativamente pelo administrador da bolsa. Sempre que o valor das ações de uma empresa muda, todos os clientes são avisados, sendo dado a conhecer o nome da empresa e no novo valor.

#### 4. Interação, com o utilizador: funcionalidades e comandos do utilizador

Nos programas **bolsa**, **cliente** e **board**, toda a interação é feita na lógica de consola: comandos escritos e informação textual sem necessidade de posicionamento de caracteres.

No programa **boardGUI**, toda a interface (input e output) é feita segundo o paradigma gráfico, usando os controlos e recursos gráficos Win32.

Todos os programas devem estar aptos a interagir com o utilizador e com os programas com os quais comunicam. Em particular, o **bolsa**, como servidor que é, deve ser capaz de lidar com vários clientes em simultâneo, o que se traduz na utilização da capacidade *multi-threaded* dos *named pipes* em Windows.

- **Acrescentar uma empresa:** comando **addc** <nome-empresa> <número-ações> <preço-ação>.

Deve indicar o nome da empresa, o número de ações disponíveis e preço inicial de cada ação. Assuma que o nome da empresa é uma única palavra. Pode assumir que não haverá mais do que 30 empresas, e se houver, pode rejeitar as que excederem esse limite.

Exemplo: *addc metroceira 500 7.50*

- **Listar todas as empresas:** comando **listc**.

Lista todas as empresas existentes na bolsa, mostrando o seu nome, número de ações disponíveis (remanescentes) e preço atual das ações.

Exemplo: *listc*

- **Redefinir custo das ações de uma empresa:** comando **stock** <nome-empresa> <preço-ação>.

Deve indicar o nome da empresa e o novo preço de cada ação.

Exemplo: *stock metroceira 9.00*

- **Listar utilizadores:** comando **users**.

Permite listar todos os utilizadores registados, mostrando o seu *username*, saldo atual e estado (atualmente ligado ou não).

Exemplo: *users*

- **Pausar as operações de compra e venda:** comando **pause** <número-segundos>.

Este comando faz com que as operações de compra e venda sejam suspensas (ignoradas) durante um período de tempo. Qualquer pedido de compra e venda que surja nesse período não será concretizado.

Exemplo: *pause 60*

- **Encerrar a plataforma:** comando **close**.

Permite encerrar o sistema. Todos os clientes e boards serão notificados, devendo terminar de seguida.

Exemplo: *close*

Assuma apenas a possibilidade de criar e listar empresas, não as pode editar, nem apagar do sistema. A única forma de remover uma empresa será reiniciar todo o sistema. Relativamente aos utilizadores, a única forma de os criar é serem lidos inicialmente do ficheiro já referido, e não será possível criar nem apagar por comando do administrador.

## Comandos – cliente

---

- **Autenticar um utilizador:** comando **login** <username > <password>.

Permite a um cliente autenticar-se junto da plataforma, utilizando o seu *username* e *password*.

Exemplo: *login maria qwerty*

- **Listar todas as empresas:** comando **listc**.

Lista todas as empresas existentes na bolsa, mostrando o seu nome, número de ações disponíveis (remanescentes) e preço atual das ações.

Exemplo: *listc*

- **Comprar ações:** comando **buy** <nome-empresa> <número-ações>.

Coloca uma operação de compra de um número de ações especificado de uma determinada empresa. Caso não exista o número de ações solicitado, a operação não deve ser concretizada, sendo o utilizador informado desse facto.

Exemplo: *buy metroceira 100*

- **Vender ações:** comando **sell** <nome-empresa> <número-ações>.

Coloca uma operação de venda de um número de ações especificado de uma determinada empresa. Se o utilizador não tiver essa quantidade de ações, a operação é rejeitada.

Exemplo: *sell metroceira 100*

- **Consultar saldo:** comando **balance**.

Permite consultar o saldo do utilizador.

Exemplo: *balance*

- **Sair da aplicação:** comando **exit**.

Sai: utilizador desliga-se do sistema e encerra o programa cliente. O sistema de bolsa continua a funcionar.

Exemplo: *exit*

## Comandos – board

---

Este programa apresenta imediatamente as informações descritas na Secção 2 e apenas deve disponibilizar ao utilizador uma forma de proceder ao seu encerramento.

## Operações – boardGUI

---

O programa boardGUI é um extra (opcional) que permite compensar partes do trabalho menos conseguidas, como especificado mais adiante na Secção de Avaliação. Toda a interação entre o utilizador e o programa boardGUI, incluindo a apresentação da informação, será obrigatoriamente realizada através de uma Interface Gráfica Win32.

O programa boardGUI deverá disponibilizar, essencialmente, a mesma informação que o programa board, mas de forma gráfica.

- Os valores das ações das empresas são apresentados como barras num gráfico de barras. A escala do gráfico e o número de empresas apresentadas são configurados através de uma *dialog box* na qual o utilizador especifica os limites inferior e superior da escala e o número de empresas N (limitado ao máximo de 10).
- O nome da empresa mais recentemente transacionada estará sempre visível na área da janela.

Deverá existir um menu que permita: redefinir os pormenores de visualização (escala do gráfico e número de empresas), sair da aplicação, e apresentar uma caixa de informações com os dados do grupo que elaborou o trabalho.

## 5. Formas de comunicação entre as aplicações

Esta especificação tem carácter obrigatório. Como já foi referido atrás:

- Bolsa – board → memória partilhada
- Bolsa – boardGUI → memória partilhada
- Bolsa – cliente → *named pipes*
- Board – boardGUI → não comunicam entre si
- Cliente – cliente → não comunicam entre si

## 6. Aspetos em aberto

Os seguintes aspetos devem ser definidos pelo aluno:

- Pormenores gráficos de visualização.
- Formato das mensagens trocadas entre os programas.
- Detalhes do modelo de dados para a gestão da bolsa de valores.
- Mecanismos de sincronização: quais e onde são necessários.
- Outros aspetos não previstos ou não explicitamente descritos.

Devem ser tomadas decisões autónomas e lógicas quanto a estes aspetos, e que não desvirtuem o sistema pretendido nem evitem os conteúdos que se pretendem ver aplicados. O sistema resultante deve ter uma forma de utilização lógica.

Ao nível do modelo de dados no servidor, devem ser tomadas as opções que simplifiquem os algoritmos dado que a primazia deste trabalho é a matéria de Sistemas Operativos 2 e não a de Programação.

## 7. Algumas chamadas de atenção

- Não coloque ponteiros em memória partilhada (pelas razões explicadas nas aulas). Isto abrange ponteiros seus e também objetos de biblioteca que contenham internamente ponteiros (por exemplo, objetos C++ STL *String*, *Vector*, etc.).
- Pode implementar o trabalho em C++, se assim quiser, desde que não oculte o API do Windows com *frameworks* de terceiros. Se o objetivo for utilizar polimorfismo, neste trabalho não irá ter grande necessidade.
- Se utilizar repositórios *git*, terá de garantir que são **privados**. Se usar um repositório público que depois seja copiado por terceiros, será considerado culpado de partilha indevida de código e terá o trabalho anulado.
- A deteção de situações de plágio leva a uma atribuição direta de 0 valores na nota do trabalho aos alunos de todos os grupos envolvidos, podendo ainda os mesmos estar sujeitos a processos disciplinares.
- Todo o código apresentado poderá ser questionado na defesa e os alunos têm obrigatoriamente de o saber explicar. **A ausência de explicação coerente é entendida como possível fraude ou como falta de conhecimento, que naturalmente se reflete na nota.**

## 8. Regras e prazos

**O trabalho é feito em grupos de 2 alunos.** Não são aceites grupos com 3 ou mais alunos. O trabalho foi desenhado e dimensionado para ser realizado em grupo de 2 alunos, sendo que, como exceção, podem eventualmente ser aceites trabalhos individuais. Todavia é aconselhada e encorajada a constituição de grupos de 2 alunos. A avaliação será realizada com os mesmos critérios independentemente dos grupos terem 2 ou, como exceção, 1 aluno.

O trabalho é composto por duas entregas: a **Meta 1** e **Meta Final**.

### Meta 1 – 13 de abril

Na entrega da **Meta 1** deve submeter apenas um relatório conciso com a arquitetura do sistema, as estruturas de dados, os esquemas e mecanismos de comunicação, os mecanismos de sincronização e descrição das situações que os tornam necessários. Este relatório não deve ultrapassar 6 páginas.

## **Meta Final – 18 de maio**

O material a entregar deverá ser:

- O trabalho completo, com os programas que constituem o sistema totalmente implementados – **bolsa**, **board** e **cliente** (eventualmente também o **boardGUI**, como extra). O projeto deve estar comprimido no formato **zip**, contendo apenas os ficheiros de controlo do projeto e de código fonte, ou seja, sem os binários (os ficheiros de *debug* das diretorias *bin* e *obj*) e sem ficheiros *precompiled headers*. Se o seu arquivo zip tiver mais de 10 Mb, poderá não conseguir fazer a submissão no Inforestudante, sendo esta responsabilidade dos alunos, que devem verificar esta questão atempadamente.
- Um relatório completo, com o máximo de 12 páginas, a explicar os pontos essenciais da implementação de cada um dos programas envolvidos, as estruturas de dados definidas e a sua utilidade, e todos os aspetos que não estejam explicitamente mencionados no enunciado e que tenham sido decididos pelos alunos, e também o diagrama com os mecanismos de comunicação e de sincronização. O relatório deve contemplar uma tabela onde indique quais os requisitos implementados, no formato:

ID	Descrição funcionalidade / requisito	Estado
...	...	implementado / não implementado (neste caso, indicar as razões)

## **9. Avaliação**

O trabalho vale **4 valores**. A nota será atribuída com base nas funcionalidades implementadas, na qualidade das soluções adotadas, na forma como são explicadas no relatório e na qualidade da defesa. O trabalho contempla uma parte opcional (o programa boardGUI) que permite não só compensar partes menos conseguidas do trabalho, como também chegar até 112,5% da nota no trabalho (4.5 valores). Relativamente à parte obrigatória do trabalho, a valorização do extra é, no máximo, **1 valor**.

A avaliação ocorre, essencialmente, na Meta Final. Na Meta 1 avalia-se o cumprimento dos objetivos estabelecidos através de uma apresentação/defesa obrigatória. Tal como descrito na ficha da unidade curricular, a avaliação será feita da seguinte forma:

- Nota da Meta 1 = percentagem entre 90% e 100%.  
A não comparência na apresentação/defesa obrigatória da Meta 1 fará com que a mesma não seja considerada. Ou seja, a percentagem obtida será a mesma que seria obtida se não entregasse a meta (90%).
- Nota da Meta Final = valor entre 0 e 100 que representa a funcionalidade e qualidade do trabalho, e a qualidade da defesa.
- Nota final do trabalho = nota obtida na Meta Final \* percentagem obtida na Meta 1.
- A Meta 1 não tem qualquer valor sem a Meta Final. Ou seja, grupos que não entreguem a Meta Final ou que não compareçam na sua defesa terão uma nota final de 0 valores.
- A falta da Meta 1 não impede a entrega da Meta Final. Apenas prejudica a nota final, já que a nota mínima da Meta 1 será automaticamente assumida (90%).

O trabalho está planeado para ser feito ao longo do semestre e a entrega do trabalho prático é única para todo o ano letivo. Não existirá trabalho prático na época especial ou noutras épocas extraordinárias a que os alunos possam ter acesso, sendo o exame sempre cotado para 14 valores.