

Trabalho prático de Sistemas Operativos

Gestão de Vendas

Grupo de Sistemas Distribuídos
Universidade do Minho

Março de 2019

1 Descrição

Neste trabalho pretende-se construir um protótipo de um sistema de gestão de inventário e vendas. O sistema deverá ser constituído por vários programas: manutenção de artigos, servidor de vendas, cliente de vendas, e agregador de dados.

1.1 Manutenção de artigos

Este programa deverá permitir inserir novos **artigos** (especificando o **nome e preço de venda**), ou alterar atributos de um dado artigo (nome ou preço). Cada artigo tem um **código numérico**, atribuído na criação como o próximo de uma sequência (1, 2, 3, ...). Para o acesso a um artigo no ficheiro ARTIGOS poder ser feito usando o código como índice, as entradas devem ter um tamanho fixo. Sendo o nome de tamanho variável, este atributo no ficheiro de artigos deve conter apenas “referências” (e.g., posição) onde se encontram num ficheiro STRINGS, em separado, contendo os respectivos nomes.

É aceitável que o ficheiro STRINGS contenha nomes obsoletos, desperdiçando algum espaço, de forma a que se possa apenas acrescentar conteúdo mesmo aquando da alteração de nomes.

Este programa recebe todo o seu input pelo seu stdin, lendo linhas de texto com o formato do exemplo seguinte:

```
$ ma
i <nome> <preço>          --> insere novo artigo, mostra o código
n <código> <novo nome>     --> altera nome do artigo
p <código> <novo preço>    --> altera preço do artigo
...
<EOF>
```

1.2 Servidor de vendas

Este programa deve controlar stocks, receber pedidos do cliente de vendas, e registar as vendas efectuadas. A quantidade em **stock** de cada artigo deverá ser mantida num único ficheiro STOCKS, para todos os artigos. Cada **venda** efectuada deverá ser registada, acrescentando uma entrada a um ficheiro VENDAS, contendo **código, quantidade e montante total da venda**.

O servidor de vendas deve ainda providenciar para correr o agregador a pedido, fazendo com que este receba o intervalo (para ser agregado) do ficheiro de vendas desde a última agregação, e fazendo com que o resultado da agregação seja escrito num ficheiro cujo nome reflecte o momento em que a agregação foi solicitada (e.g., 2019-03-29T14:23:56).

1.3 Cliente de vendas

O cliente de vendas deve interagir com o servidor de vendas, solicitando-lhe a execução de (uma sequência de) operações que se distinguem facilmente pelo número de parâmetros introduzidos. Uma das operações retorna a quantidade em stock e o preço de um artigo (identificado pelo código). A outra operação permite efectuar vendas ou entrada em stock, especificando o código e quantidade (negativa ou positiva, respectivamente):

```
$ cv
<código_numérico>          --> mostra no stdout stock e preço
<código_numérico> <quantidade> --> actualiza stock e mostra novo stock
...
<EOF>
```

O sistema deverá permitir a execução concorrente de vários clientes de vendas. Teste o seu trabalho com dois ou mais clientes a "martelarem" simultaneamente o mesmo artigo!

A interação dos utilizadores com este cliente de vendas é feita apenas pelo *stdin*, sendo os resultados retornados no *stdout*. Por favor não invente, não escreva mais nenhuma mensagem para o *stdout*. Inicialmente poderá querer testar manualmente o servidor de vendas mas depois não se esqueça de criar um ficheiro contendo (muitos, centenas ou milhares de) comandos de compra e venda de um artigo, lançar uma dúzia de clientes que lêem esse ficheiro e verificar pelo stock final se alguma operação se "perdeu".

1.4 Agregador

O programa agregador funciona como filtro. Recebe pelo *stdin* entradas no formato do ficheiro de vendas, até end-of-file. Nessa altura produz para o *stdout* os dados agregados de cada artigo com vendas efectuadas, contendo o código do artigo, a quantidade total e o montante total de vendas do artigo respectivo (mantendo o formato do ficheiro de vendas).

2 Alguns aspectos valorizados

2.1 Caching de preços

No servidor de vendas, para evitar o acesso constante ao ficheiro ARTIGOS, e tendo em conta que vários artigos populares serão continuamente requisitados, mantenha em memória a informação de preços que foi lida deste ficheiro, tendo cuidado com a sua invalidação por uma alteração de preços. Ou seja, quando são efectuadas alterações a preços de artigos, tal deve comunicado ao servidor de vendas, para este poder usar os preços actualizados.

2.2 Agregação concorrente

Para permitir explorar agregação concorrente, faça com que o servidor de vendas ponha a correr várias instâncias do programa agregador, dividindo o intervalo a agregar por estas, e fazendo com que os resultados das sub-agregações sejam por sua vez agregados.

2.3 Compactação do ficheiro STRINGS

Para reduzir o desperdício de espaço, quando este superar os 20% do ficheiro, deverá ser feita a compactação para um novo ficheiro e ajustadas as posições no ficheiro de artigos. Esta operação deverá ser a última a acrescentar ao seu trabalho.

Lembre-se que "a verdade" está nos ficheiros e que esses ficheiros poderão ter dimensão "muito grande". Em memória estará a cache. Isto facilitará a recuperação do serviço após um crash (ou *kill*) do servidor.

Informações gerais

- Conforme indicado no início do ano, o método de avaliação de Sistemas Operativos contempla a realização e discussão de um trabalho prático onde é necessário ter nota mínima de 10 valores. Quem teve positiva no trabalho do ano imediatamente anterior pode optar por manter a nota (truncada a 15). Para "descongelar" a nota, basta que não se inscreva em nenhum grupo TP deste ano¹.
- Cada grupo de trabalho deve ser constituído por até três elementos. Grupos de dimensão inferior a três terão de ser justificados e autorizados pelos docentes.
- Serão disponibilizados no blackboard muitos grupos (LCC 25, MIEI 100) com capacidade para 3 elementos. Quando o grupo estiver formado, APENAS UM dos elementos escolhe um grupo que esteja vazio e inscreve-se nesse grupo. De seguida comunica aos restantes elementos o número do grupo em que se inscreveu para que os seus colegas se registem também nesse grupo. Note que as inscrições terminam logo após as férias da Páscoa, cerca de 15 dias antes da entrega do trabalho.
- O trabalho deve ser entregue até às 23:59 do dia 11 de Maio, através do Blackboard. Deve ser enviado um único ficheiro em formato ZIP, contendo um pequeno relatório em PDF, o código fonte, a makefile para compilação de todo o código desenvolvido. Será essa a versão que irão demonstrar no dia da discussão.
- A avaliação dos trabalhos deverá ocorrer nos dias 13 a 17 de Maio. Imediatamente após contagem do número de trabalhos submetidos será publicado no blackboard um conjunto de horários (em princípio 20 minutos por grupo). Cada um destes grupos do Blackboard corresponderá a um determinado slot, sendo necessária a presença de TODOS os elementos desse grupo de alunos
- Para evitar percalços e atrasos no setup da demonstração do trabalho, esta poderá ser pedida a qualquer dos elementos do grupo enquanto outro vai explicando as decisões tomadas. Poderá ser realizada num computador qualquer, seja de um dos profs ou colega de grupo; toda a gente tem de saber instalar e correr o software sem comprometer o horário de avaliação.
- Para permitir a automatização do teste e avaliação dos trabalhos, **é obrigatório uniformizar os nomes dos ficheiros de dados, programas e formatos de input:**
 - ma (manutenção de artigos)
 - sv (servidor de vendas)
 - cv (cliente de vendas)
 - ag (agregador)
 - artigos, strings, stocks, vendas

¹Confirme que o seu nome/nota figura na lista de notas "congeladas", a publicar quando forem abertas as inscrições.