



PCS 3111 - Laboratório de Programação Orientada a Objetos para Engenharia Elétrica

2023

Aula 10 – Persistência de Objetos

Atenção

- Código inicial a ser usado na resolução dos exercícios encontra-se **disponível no e-Disciplinas**.
- Submeta um arquivo comprimido (faça um “.zip” – **não pode ser “.rar”**) colocando apenas os arquivos “.cpp” e “.h”. Não crie pastas no “zip”.
- Os nomes, os atributos, os métodos, e as respectivas assinaturas das classes dadas **devem seguir o especificado** em cada exercício para fins de correção automática.

Exercício 1

Implemente uma nova classe, **PersistenciaDeProduto**, que será responsável pela persistência dos objetos da classe **Produto** entregue (e que não deve ser alterada). Essa classe possui os seguintes métodos:

```
class PersistenciaDeProduto {  
public:  
    PersistenciaDeProduto(string arquivo);  
    virtual ~PersistenciaDeProduto();  
    void inserir(Produto *produto);  
};
```

- O construtor de **PersistenciaDeProduto** recebe uma string com o nome do arquivo em que os dados serão salvos. O destrutor não precisa fazer nada.
- O método **inserir** deve inserir no arquivo informado no construtor os dados do **Produto** passado como argumento, adicionando-o ao final do arquivo.

O arquivo a ser usado por **PersistenciaDeProduto** deve possuir o seguinte formato:

```
nome1  
preco1  
nome2  
preco2  
...  
nomen  
precon
```

No arquivo serão armazenados os dados (nome, preço) de **n Produtos**. A primeira linha do arquivo deverá conter o nome do primeiro **Produto**, a segunda linha deve conter o seu preço. A



partir da terceira linha esse padrão se repete para o segundo **Produto** e assim por diante para os n próximos Produtos. **Note que o arquivo termina com um “\n” (use o endl)!**

Exemplo: Sejam dois **Produtos**, “Lasanha” e “Batata”, cujos preços sejam, respectivamente, 20 e 5 reais. Teremos um arquivo texto com a seguinte apresentação (em um editor de texto):

```
Lasanha
20
Batata
5
```

Execute os seguintes passos na função teste1:

1. Crie dois Produtos: “Lasanha” de 20 reais e “Batata” de 5 reais;
2. Crie também um objeto **PersistenciaDeProduto** que deve persistir os dados no arquivo teste1.txt;
3. Insira os dois Produtos criados, primeiro “Lasanha” e depois “Batata”.
4. Delete todos os objetos alocados dinamicamente.

Exercício 2

Adicione à classe **PersistenciaDeProduto** o método

```
Pedido* obter();
```

que retorna um novo objeto **Pedido** (classe já fornecida e que não deve ser alterada) com os **Produtos** armazenados no arquivo informado no construtor da classe. Por simplicidade, considere que o arquivo contém no máximo 10 **Produtos**. Desse modo, deve-se reconstruir todos os **Produtos** armazenados no arquivo, seguindo o formato explicado no exercício anterior.

Assume-se que quem chamou o método será o responsável por destruir o **Pedido**.

Esse método deve ser implementado de modo a lidar com possíveis problemas de leitura e formatação do arquivo:

- Caso o arquivo não seja encontrado, o método deve jogar a exceção **invalid_argument**, de argumento “Erro de leitura”, da biblioteca padrão.
- Caso o arquivo esteja vazio, a função deve retornar **NULL**.
- Caso o arquivo não siga a formatação esperada ou haja uma falha de leitura, deve-se jogar a exceção **logic_error**, de argumento “Arquivo com formatacao inesperada”, da biblioteca padrão.

DICA: siga o estilo apresentado no slide 27 do material de aula. Ou seja, antes do while obtenha do arquivo os valores esperados para a primeira leitura em variáveis temporárias. Então faça um while verificando a entrada. Dentro do while use as variáveis temporárias para o que for necessário. A última parte do laço deve obter novamente do arquivo o valor das variáveis temporárias.



ATENÇÃO: Assuma que o nome do **Produto** não contém espaços!

Complete a função teste2 com os seguintes passos na função:

1. Crie um objeto da classe **PersistenciaDeProduto**, passando o nome *teste2.txt* (o arquivo teste2.txt é fornecido, mas o Judge usa um outro) como argumento;
2. Em um bloco try, faça:
 - a. Pelo método obter, obtenha o Pedido referente aos Produtos do arquivo *teste2.txt*;
 - b. Chame o método imprimir do Pedido, se for diferente de NULL
 - c. Delete o Pedido
3. No catch capture a eventual exceção, imprima na saída padrão o retorno do método `what()` (pulando uma linha ao final) e, então, delete-a.
4. Delete o objeto de persistência.

Testes do Judge

Exercício 1

- PersistenciaProduto inserir um Produto em arquivo vazio
- PersistenciaProduto inserir mais de um Produto em arquivo vazio
- PersistenciaProduto inserir Produtos em arquivo não vazio
- Teste da função teste1

Exercício 2

- PersistenciaProduto obter para arquivo vazio
- PersistenciaProduto obter para arquivo mal formatado
- PersistenciaProduto obter para arquivo inexistente
- PersistenciaProduto obter em arquivo com um Produto
- PersistenciaProduto obter em arquivo com mais de um Produto
- Teste da função teste2