



Universidade Federal de Pelotas
Centro de Desenvolvimento Tecnológico
Cursos de Ciência da Computação e Engenharia de
Computação

Aula Prática 01 – 2021/1

Implementação de um Bloco de Lembretes

Exercício 1: Crie uma classe chamada *Lembrete*. O objetivo desta classe é guardar os seguintes dados de um lembrete: data para a notificação do lembrete (Lembre da abstração e reusabilidade de código!), nome do lembrete e descrição. Defina pelo menos um construtor de inicialização e um método para imprimir os dados do lembrete.

Exercício 2: Crie uma classe chamada *BlocoDeLembretes*. O objetivo desta classe é guardar um conjunto de lembretes, modelados pela classe *Lembrete*, em um array, implementando uma estrutura de lista (lista sequencial). Ao instanciar um array, é necessário determinar o seu tamanho, que neste exercício deve ser de 10 posições. Nos próximos exercícios você adicionará mais recursos à estrutura da classe. Você pode implementar o método *main* para testar as funcionalidades implementadas.

Exercício 3: Adicione um método chamado *incluirLembrete*, que recebe um objeto *Lembrete* como parâmetro e o adiciona no array, sempre na última posição da lista (lembre a última posição da lista não é necessariamente a última posição do array).

Exercício 3: Adicione um método chamado *numeroDeLembretes*, que retorna o número de lembretes armazenados no array.

Exercício 4: Adicione um método chamado *removerLembrete*, que recebe o número (índice/posição no array) do lembrete a ser removido e o exclui da lista.

Exercício 5: Adicione um método chamado *listarLembretes* para imprimir todos os lembretes armazenados na lista.

Exercício 6: Adicione um método chamado *buscarLembretesPorData* para listar todos os lembretes da lista que ocorrem na mesma data passada como parâmetro no método.

Exercício 7: Adicione um método chamado *buscarLembretesPorNome* para listar todos os lembretes da lista que contenham a *substring* que será passada como parâmetro do método em seu nome.

Exercício 8: Altere o método *incluirLembrete* para que a inclusão de novos lembretes na lista não fique limitada ao valor inicial atribuído com tamanho do array. Ao invocar o método de inclusão, se não houver mais espaço no array, deve-se (1) instanciar um novo array com o dobro do tamanho do array atual, (2) copiar os dados do array atual para o

novo e (3) incluir o novo lembrete no novo array. Lembre de (4) atualizar o atributo da classe *BlocoDeLembretes* que referencia o array atual para o novo array instanciado.

Exercício 9: Crie uma nova classe do projeto, que deve se chamar *LembretesApp*. Esta classe será a classe principal do projeto e deve prover métodos para entrada e saída de dados relativos a um bloco de lembretes. A princípio, ela deve conter apenas um atributo para guardar um objeto do tipo *BlocoDeLembretes*.

Exercício 10: Adicione o método *menu* à classe *LembretesApp*. Ele deve imprimir as mensagens abaixo e ser capaz de ler a opção escolhida pelo usuário, retornando um valor inteiro através do método. Observe que 'XX' deve ser substituído pelo número de anotações armazenados no objeto *BlocoDeLembretes*.

```
Lembretes App - A lista contém XX lembretes.  
Escolha uma das seguintes opções:  
1- Adicionar novo lembrete  
2- Remover lembrete  
3- Listar  
4- Busca por data  
5- Busca por nome  
6- Sair
```

Exercício 11: Adicione o método *main* à classe *LembretesApp*. Este método deve instanciar um objeto da classe *LembretesApp* e deve criar um laço para chamar o método *menu* do objeto instanciado. Dependendo da opção escolhida, ele deve invocar os respectivos métodos para realizar as operações sobre a lista de lembretes. O laço principal do *main* deve cessar quando o usuário escolher a opção 'Sair'.