

## PILHA – Exercícios – Continuação

### Estrutura de Dados e Armazenamento

- 4- Implementar um programa que utilize uma pilha para ler uma frase e exibi-la invertida. Neste caso, o elemento da pilha deve ser do tipo Character (use a PilhaObj).

Ex: a frase "A pilha do gato" deve sair "otag od ahlip A"

Dica Java: para trabalhar com cada caractere da String, pode-se obter cada caractere da String, utilizando o método charAt(índice)

```
Ex: for (int i=0; i < frase.length( ); i++) {  
    System.out.println (frase.charAt(i));  
}
```

- 5- Implementar um programa que utilize uma pilha para verificar se uma dada cadeia de caracteres é ou não palíndroma. Uma cadeia de caracteres é palíndroma quando tem a mesma leitura, lendo-se da esquerda para direita ou da direita para esquerda (sem considerar os espaços em branco, tracinho, acento, letra maiúscula/minúscula).

Por exemplo: radar, ovo, Ana, subi no ônibus, Socorram-me subi no ônibus em Marrocos, anotaram a data da maratona, a torre da derrota

- 6- Crie um projeto para este exercício.

Nesse projeto, crie uma classe chamada Livro, com atributos nome e autor. Implemente os getters e setters, o construtor e o toString().

Copie a classe PilhaObj para este projeto.  
Implemente o método a seguir na classe PilhaObj:

```
public T popElementoDoIndice (int n)  
// esse método retorna o elemento da pilha que está no índice n,  
// contando a partir do topo.  
// ou seja, o topo é o índice zero, topo -1 é o índice 1  
// e assim sucessivamente  
// mas a pilha deve permanecer com os elementos sem o elemento retirado  
// se n for maior do que o número de elementos da pilha, retorna null  
// senão cria uma pilha aux (objeto da classe PilhaObj <T>) e empilha em  
// aux os n elementos desempilhados da pilha (um de cada vez)  
// salve numa variável chamada alvo o pop do próximo, que é o elemento alvo  
// depois devolva os elementos desempilhados para a pilha original,  
// ou seja, enquanto a pilha aux não está vazia, desempilhe um elemento  
// de aux e empilhe esse elemento na pilha original  
// retorne o elemento alvo
```

Nesse mesmo projeto, crie uma classe chamada Teste, e configure para que ela tenha o método main

No main:

- crie um objeto da classe PilhaObj <Livro>, chamado pilha
  - crie vários objetos da classe Livro e empilhe-os, um a um, na pilha.
  - exiba a pilha
- 
- Simule a situação em que você deseja tirar um livro da pilha, mas ele não está no topo da pilha. Então, você deve usar o método acima popElementoDoIndice, para “tirar” o livro desejado da pilha, mantendo os demais livros empilhados como inicialmente.