

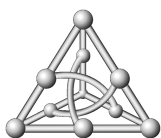
## Prova Optativa 1

Algoritmos e Programação Orientada a Objetos II — 2018

### Instruções para a realização da prova:

1. A prova contém **2 questões**, totalizando **10 pontos**;
2. A prova tem duração de **1 hora e 50 minutos**;
3. **Não altere a disposição do mobiliário da prova**;
4. **Coloque seu celular** em cima do gabinete, desligado. Caso o professor ouça o celular vibrando ou tocando durante a aula, o aluno receberá **nota zero**;
5. **Faça a prova em silêncio**; não converse durante a prova;
6. **Não tente plagiar a prova do(a) seu(sua) colega**, você pode prejudicar você e seu(sua) colega;
7. **Utilize apenas o que foi ensinado em sala de aula**. O uso de qualquer estrutura de programação ou estrutura de dados que não foi ensinada em sala de aula anulará a sua questão.
8. Para esta prova, ainda não é permitido utilizar as coleções genéricas do Java (*ArrayList*, *LinkedList*, etc.);
9. Utilize a IDE ou editor de texto de sua preferência;
10. Crie uma pasta com o seu nome dentro do diretório *Documentos*, e coloque o seu projeto dentro desta pasta.

**BOA PROVA!**



### Questões

1. [5.00] Nesse exercício você demonstrará conhecimento sobre listas lineares e os algoritmos de ordenação elementares.
  - (a) Em uma classe chamada *ListaLinearEncadeadaComCabeca*, implemente uma lista linear encadeada com cabeça (de inteiros), que deve prover as seguintes operações:
    - i. Inserção de um elemento no final da lista;
    - ii. Retornar a célula em uma determinada posição  $i$  ( $i$ -ésima célula). Assuma que o valor  $i$  passado sempre corresponderá a uma posição válida da lista;
    - iii. Trocar o conteúdo da célula em uma posição  $i$  ( $i$ -ésima célula) com o conteúdo da célula de elemento de uma posição  $j$  ( $j$ -ésima célula). Assuma que as posições  $i$  e  $j$  são válidas;
    - iv. Retornar o tamanho da lista.
  - (b) Crie uma classe chamada *Ordenacao*, que contém um método *ordena*, que recebe uma lista linear encadeada por parâmetro (utilizando um objeto do mesmo tipo criado no item *a*) e ordena a lista em ordem crescente utilizando um dos algoritmos de ordenação elementares vistos em sala de aula.

*DICA: você não precisa repensar todo o algoritmo de ordenação pelo fato de estar usando uma lista linear encadeada.*
  - (c) Crie uma classe chamada *Exercicio1*, que contém um método *main* que testa as classes dos itens *a* e *b*.
2. [5.00] Uma possível maneira de criptografar uma mensagem é inverter cada palavra do texto. Por exemplo, o texto "Uma mensagem confidencial", quando criptografado, fica "amU megasnem laicnedifnoc".

Implemente a criptografia de uma mensagem (com no máximo 100 caracteres) digitada pelo usuário **utilizando uma Pilha**.

Nesse exercício, você deve criar pelo menos as classes Pilha e TesteCriptografia (com o método main).