

**Atividade Acadêmica:** Algoritmos e Programação – Estruturas Lineares

**Professor:** Guilherme Silva de Lacerda (guilhermeslacerda@gmail.com – gslacerda@unisinos.br)

## Laboratório 6

### Importante! Regras a serem seguidas

- 1) Todo e qualquer código deve ter seu teste equivalente (classe para testar o código)

**Exemplo: Classe Conta tem que ter uma ContaTeste que realize os testes nela**

- 2) Respeite as convenções de código

(<https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>)

- 3) Estruture o projeto com pacotes, conforme exemplo

**Nome do Projeto: Laboratorio6\_Eleicoes**

**pacote1\**

*Classe1*

*Classe2*

**pacote2\**

*Classe3*

*Classe4*

#### Importante:

- **Usar testes unitários automatizados**
- Atividade desenvolvida em duplas
- Atividade a ser entregue no ambiente EAD
- Formato de entrega: Arquivo ZIP dos códigos (sugestão de nome: **Lab6\_NomeAluno1\_NomeAluno2.zip**)
- Implementar testes automatizados com JUnit

### Enunciado

Desenvolver uma aplicação Java que receba as seguintes informações de candidatos em uma eleição: nome do candidato, partido político, idade, número de votos.

Crie diferentes comparadores para ordenação (número de votos, idade, partido político e nome).

Ao final, o programa deve apresentar as seguintes informações:

- Candidato mais jovem
- Candidato mais velho
- Candidato mais votado
- Candidato menos votado
- Total de votos recebidos por todos os candidatos
- Média de votação recebida pelos candidatos

Use a *API Collections* para reestruturar a coleção para armazenar os candidatos e fazer a computação.

Crie testes unitários automatizados. Crie cenários para estas situações.

É possível criar um menu de opções para que o usuário consiga acessar as informações.