

PRONTUÁRIO: INDICAR NO ZIP

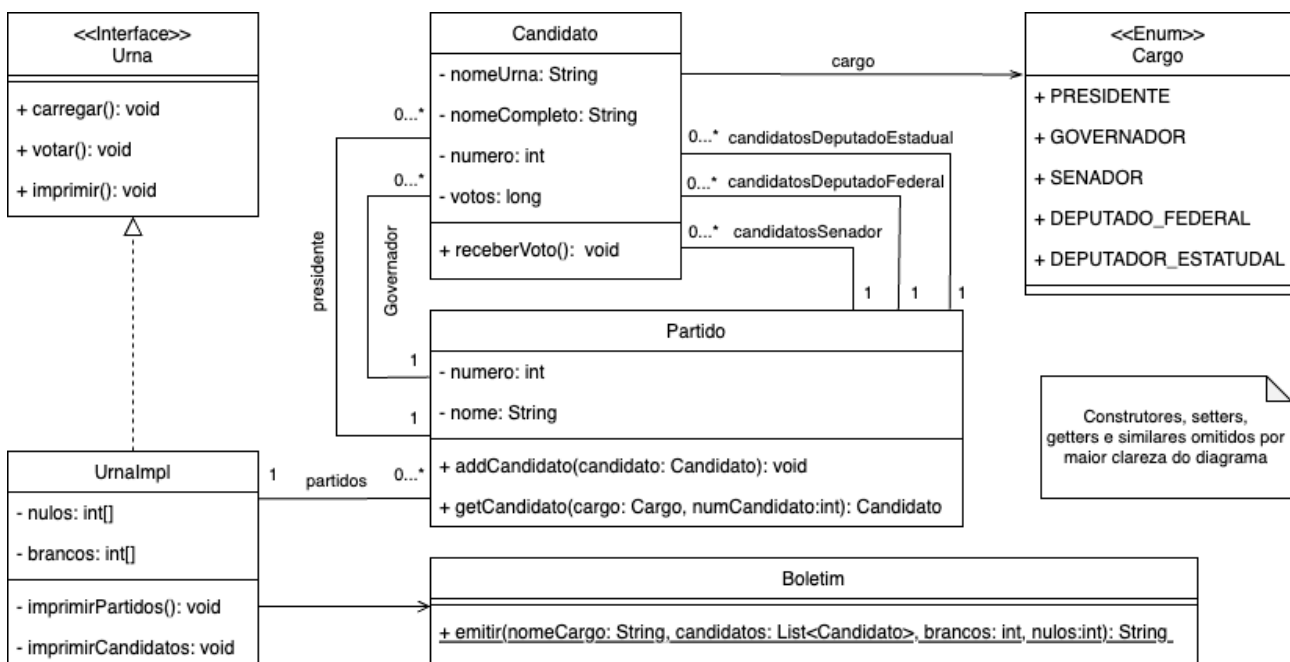
Dia: 13/05/2025

Horário: 8h00 – 11h30

<ul style="list-style-type: none"> <li>✓ A prova é individual e sem consulta.</li> <li>✓ Não é permitido utilizar qualquer código não fornecido junto com a prova.</li> <li>✓ Não coloque nome na prova ou projeto.</li> <li>✓ Atribui-se nota zero à prova em desacordo com o item acima.</li> </ul>	<ul style="list-style-type: none"> <li>✓ A prova deve ser nomeada da seguinte forma: PRONTUARIO_P2, com o "SC".</li> <li>✓ Não envie apenas as classes, mas todo o projeto.</li> <li>✓ Envie o projeto como zip no Moodle.</li> </ul>
---	---

*"Democracy is the worst form of government, except for all the others."* WINSTON CHURCHILL

Em 2026, o Brasil realizará sua décima sétima eleição desde a redemocratização. Quatorze dessas eleições foram realizadas com o suporte de um sistema eletrônico, que permite rápida apuração, garante o sigilo do voto e mitiga más práticas eleitorais, como o voto de cabresto e a destruição de cédulas. Pensando nisso, hoje você fará seu próprio sistema de urna eletrônica, muito menos seguro e testado que o nosso, claro. Esse sistema terá dados reais de candidaturas realizadas no Tribunal Superior Eleitoral – TSE, por meio de um dataset público contendo informações de candidatos e partidos. Um membro de sua equipe já implementou uma classe chamada *Utilitario*, que possui o método *carregarCandidatos()*, responsável por retornar uma lista com todas as candidaturas disponíveis no arquivo, com os dados de cada candidatura separados por “;”. Como você é um desenvolvedor que conhece orientação a objetos, não trabalhará apenas com dados brutos, mas com classes de objetos e suas relações, conforme o diagrama a seguir.





Quando a urna for criada, os dados do arquivo deverão ser carregados na memória, particionados e utilizados para criar os objetos do modelo, representando candidatos e partidos, de forma a permitir o exercício da democracia. Com os candidatos já disponíveis na urna, o sistema deve permitir que diversos eleitores registrem seus votos, na ordem definida pelo TSE: Deputado Federal, Deputado Estadual, Senador, Governador e Presidente. O sistema deve permitir que o eleitor digite os números de cada candidato ou BRANCO. Após a digitação, o sistema deve exibir as informações do candidato e as opções CORRIGE e CONFIRMA. Encerrado o processo de votação, o

sistema deverá gerar cinco boletins de urna, um para cada tipo de cargo em disputa, pela ordem. Em cada boletim deverá constar os candidatos votados e seus respectivos votos, o número de votos brancos, nulos e o total.

Execute as atividades a seguir para a implementação do exercício em Java. Você pode criar outros métodos, principalmente privados, para melhor organizar sua aplicação. Para a atribuição da nota será levada em conta não apenas a funcionalidade, mas a qualidade, adequação e pertinência de cada solução. Bom senso faz parte da prova.

#	Descrição	Pont.
1	Crie as classes do modelo utilizando os modificadores de acesso mais adequados aos métodos e atributos, segundo o conceito de encapsulamento. Crie os construtores que achar mais adequados. Implemente o Cargo como um Enum.	0,5pt
2	Estruture as associações de composição das classes do modelo. Como há milhares de candidatos, para relacionamentos do tipo 1 para 0..*, utilize mapas ao invés de listas ou arrays.	0,5pt
3	Na classe Partido, crie um método <i>addCandidato(Candidato candidato)</i> que recebe um candidato e o armazena em sua respectiva estrutura de dados, a depender de seu cargo. Crie também um método <i>getCandidato(Cargo cargo, int numCandidato)</i> , que obtém um dado candidato a partir dos parâmetros informados.	1,0pt
4	Implemente os métodos <i>equals()</i> e <i>hashCode()</i> , levando em conta os atributos que tornam um Candidato e um Partido únicos no modelo.	0,5pt
5	Crie uma interface chamada Urna, que contenha os métodos previstos no diagrama do modelo da aplicação.	0,5pt
6	Crie uma classe chamada UrnaImpl, que forneça uma implementação concreta para a interface Urna. No construtor, chame o método <i>carregar()</i> .	0,5pt
7	Implemente o método <i>carregar()</i> da classe <i>UrnaImpl</i> . Use o método <i>carregarCandidatos()</i> da classe <i>Utilitario</i> para obter a lista de candidatos. Para cada linha obtida, extraia as informações para a construção de objetos Partido (quando necessário) e Candidato. Estabeleça o relacionamento entre os objetos e armazene no mapa <i>partidos</i> . Dica 1: para extrair dados da String, use o método <i>split()</i> . Dica 2: O número do partido aparece nos dois primeiros dígitos do número do candidato, por isso, use o método <i>substring()</i> . Ex: O candidato de número 99232 pertence ao partido de número 99.	1,5pt
8	Implemente os métodos privados <i>imprimirPartidos()</i> e <i>imprimirCandidatos()</i> , que devem ser utilizados para imprimir no console todos os objetos criados para os partidos e candidatos, nos formatos disponíveis nos Quadros A e B. Invoque esses métodos ao final do método <i>carregar()</i> implementado no item anterior, para comprovar que todos os candidatos podem ser votados. Dica: para ajudar no alinhamento, você pode utilizar o método <i>textoFixo()</i> disponível na classe <i>Utilitario</i> .	0,5pt

9	Implemente o método <i>votar()</i> , que realiza a votação completa de um eleitor, na ordem de votação prevista pelo TSE. Para cada voto, o sistema deve: i) informar o cargo e as opções; ii) exibir as informações do candidato, VOTO BRANCO ou VOTO NULO, a depender do informado pelo eleitor; e iii) após digitar CONFIRMA, contabilizar o voto para o candidato, incrementar VOTO BRANCO ou incrementar VOTO NULO. VOTO NULO é o voto no qual o número informado não corresponde a nenhum candidato ao cargo. Veja o formato de entrada no Quadro C.	2,5pt
10	Implemente o método <i>imprimirBoletim()</i> , que consolida os votos de cada cargo e invoca o método <i>emitir()</i> da classe utilitária Boletim, a ser implementada por você. Por ser utilitária, a classe Boletim não deve ser instanciável ou herdável, sendo o <i>emitir()</i> oferecido como método de classe. Utilize um StringBuilder para montar o boletim de urna. O extrato a ser gerado para cada cargo em votação deve ter o formato do Quadro D.	1,5pt
11	Crie uma classe Principal, que contenha um método main. Nessa classe, crie uma instância de Urna, repita o processo de votação algumas vezes e imprima os boletins de urna com os resultados. Para controlar a votação, utilize um loop e uma constante para o número de eleitores que irão votar.	0,5pt
	Implementar métodos demasiadamente ineficientes do ponto de vista computacional.	-1,0pt
	Não seguir as orientações sobre a criação e entrega do projeto descritas no preâmbulo da prova.	-1,0pt

**Prêmio Usain Bolt:** O aluno que terminar todas as atividades corretamente primeiro ganha 1pt adicional para usar na segunda prova. Você está voando?

*"O preço a pagar pela tua não participação na política é seres governado por quem é inferior"*  
PLATÃO (428 a.C. - 348 a.C.)

O seu voto (ou a ausência dele) impacta a forma como vivemos, nossa Saúde, Ciência e Educação. Sempre estude o histórico dos candidatos para saber o fizeram ou deixaram de fazer pelo SUS, pela Educação Pública e pela Ciência no Brasil. Comece pela história do próprio IFSP ...

**\*\*\* Boa sorte! \*\*\***

Quadro 1 – Exemplo de Listagem de Partidos

-----PARTIDOS-----	
Número	Nome
10	REPUBLICANOS
11	PP
12	PDT
13	PT
...	

Quadro 2 – Exemplo de Listagem de Candidatos

----- CANDIDATOS A DEPUTADO ESTADUAL -----			
Nome	Cargo	Número	Partido
JORGE CUTIGI	DEPUTADOR ESTATUDAL	20369	BES
LUCAS BUENO	DEPUTADOR ESTATUDAL	10114	P00
...			

Quadro 3 – Exemplo de Interface de Votação

```

Deputado Federal [4 números | BRANCO]: 9908
Nome: SANTINHO                               Partido: UMAB (99)
[CORRIGE | CONFIRMA]: CONFIRMA

Deputado Estadual [5 números | BRANCO]: BRANCO
VOTO EM BRANCO
[CORRIGE | CONFIRMA]: CONFIRMA

Senador [3 números | BRANCO]: 138
VOTO NULO
[CORRIGE | CONFIRMA]: CORRIGE
...

```

Quadro 4 – Exemplo de Boletim de Urna

===== JUSTIÇA ELEITORAL =====		
Boletim de Urna		
-----PRESIDENTE-----		
Nome	Número	Votos
FABINHO	77	43
CARLÃO	88	68
-----		
Total de votos Nominais: 111		
Branco: 3		
Nulos: 2		
Total apurado: 106		
-----		