

TRABALHO PRÁTICO 1

Data de entrega: 15/10

Competição em: 17/10

O BUROCRATA

Certa vez no Alegrete, o sistema de processos acadêmicos da UNIPAMPA sofreu com uma pane global que inviabilizou seu uso por várias semanas. Como as atividades da universidade não podem parar, um servidor público recebeu a tarefa de processar todos os documentos dos cursos de graduação e pós-graduação do campus, recebendo, classificando e dando o encaminhamento para todos os processos. Coincidentemente, esse servidor era obcecado pela formalidade de documentos, pelo cumprimento das regras e pela beleza dos procedimentos administrativos: ele era um **burocrata**. Essa pessoa teria que receber documentos e encaminhá-los em processos de forma otimizada e, ao mesmo tempo, respeitar as regras administrativas da universidade.

Esse burocrata precisa atender simultaneamente dez cursos da universidade, recebendo os documentos e despachando esses documentos em processos. O burocrata pode pegar um documento de um monte de um curso (que não está em qualquer ordem particular) e colocar em um processo que esteja aberto em sua mesa (que pode manter cinco processos abertos simultaneamente). Os documentos em um processo também não têm qualquer ordem em particular. Esse burocrata pode também despachar um processo para a secretaria acadêmica. Uma nova pasta (processo vazio) será aberta por outro funcionário na mesa do burocrata assim que possível, mas não imediatamente após um despacho. Os montes com documentos de um curso podem acabar ficando vazios em algum momento, mas podem também receber novos documentos a qualquer instante.

As regras de organização e despacho de processos precisam ser cumpridas e visam não misturar certos tipos de documentos, viabilizar análises jurídicas e encaminhar os processos para departamentos corretos. Considere as seguintes regras:

- (1) Um processo não pode conter Documentos de cursos de graduação e cursos de pós-graduação ao mesmo tempo quando for despachado.
- (2) Um processo não pode misturar Documentos Administrativos e Documentos Acadêmicos, mas atas podem estar em qualquer processo. Assim, uma Norma não pode ser despachada junto de um Histórico no mesmo processo, por exemplo.
- (3) Um processo não pode ser despachado apenas com Atas.
- (4) Uma Portaria ou um Edital com 100 ou mais páginas é um “documento substancial” e deve ser despachado em um processo sem qualquer outro documento junto. Contudo, Portarias e Editais que não sejam mais válidos podem ir junto de outros documentos mesmo que sejam substanciais.

(5) Diferentes Circulares e Ofícios só podem ser despachados no mesmo processo se tiverem um destinatário em comum. Por exemplo, um Ofício para “Ana Moura”, uma Circular para “Dulce Pontes” e “Ana Moura”, e uma Circular para “Ana Moura”, “Antônio Variações” e “Amália Rodrigues” podem ser despachados juntos no mesmo processo, porque Ana Moura é o destinatário comum entre todos.

(6) Diplomas só podem ser despachados junto de outros Diplomas, Certificados ou Atas.

(7) Atestados de diferentes categorias não podem estar em um mesmo processo.

Descumprir essas regras causa o aumento do estresse do burocrata.

A mesa do burocrata só pode ter cinco processos abertos para colocação de documentos. Ainda, uma informação de extrema importância é que cada processo suporta, no máximo, 250 páginas. Colocar páginas acima do limite causa rompimento da pasta do processo durante o transporte (depois que ela é despachada para a secretaria acadêmica) e perda de todos os documentos envolvidos, causando um aumento substancial de estresse do burocrata que recebe uma advertência administrativa.

Sempre que o burocrata despacha um processo para a secretaria acadêmica, um novo processo vazio é criado no lugar daquele que foi enviado.

Além desse documento, estão disponíveis no ambiente Moodle:

1. O código fonte para o simulador da Universidade, e
2. Arquivos HTML contendo as especificações detalhadas das classes em JavaDoc.

Uso de Inteligência Artificial nessa atividade: autorizado com restrições

Você pode usar recursos de Inteligência Artificial (IA) para ajudar na construção do trabalho, mas deve evitar usar códigos prontos fornecidos diretamente pela IA para partes críticas como métodos equals, hashCode, hierarquia das classes e lógica do burocrata.

Qualquer código gerado por IA que seja usado sem modificações ou com “modificações não significativas” deve ser indicado por meio de comentários. Por exemplo:

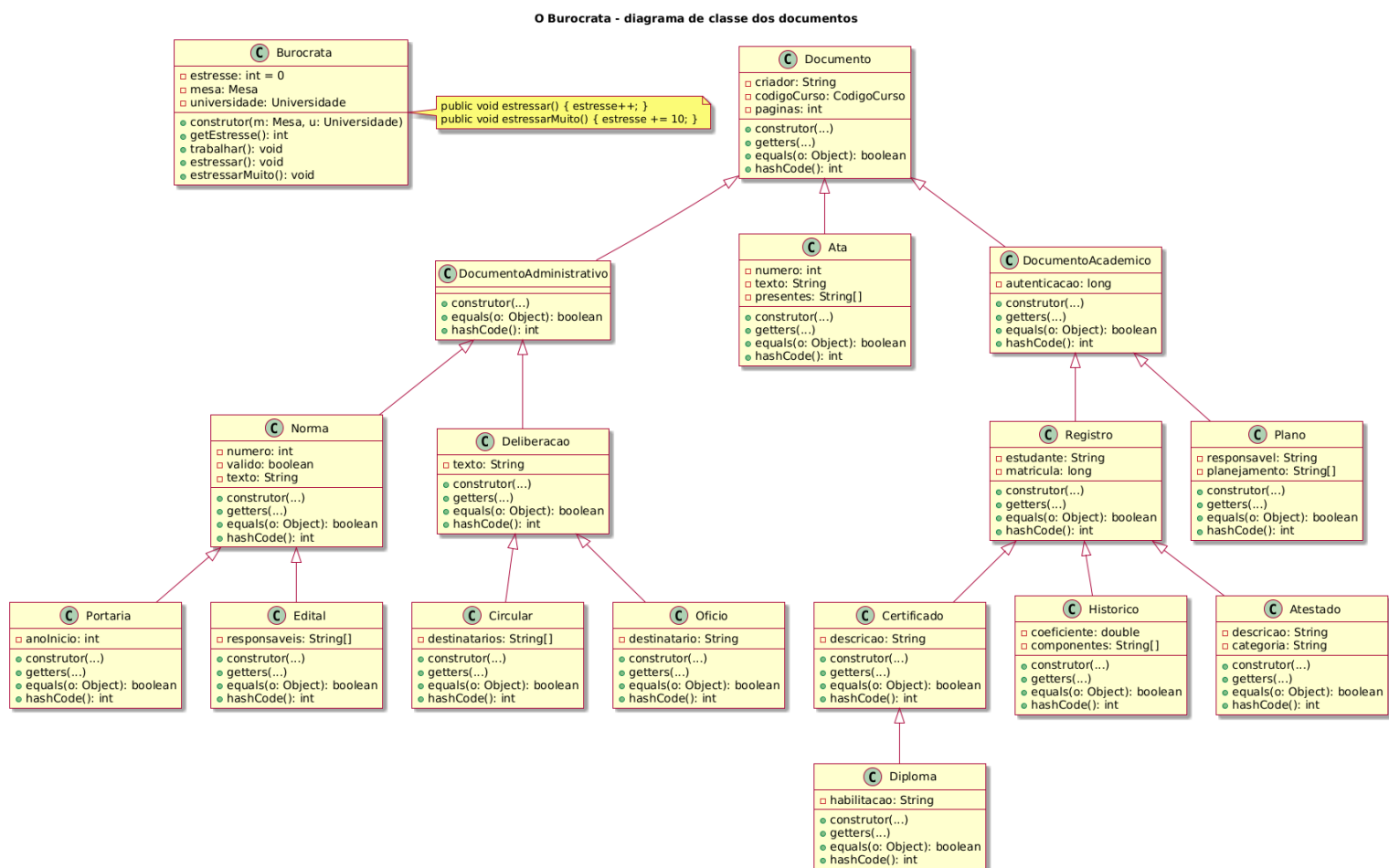
```
/* começo de código gerado por IA */  
int significadoDaVidaDoUniversoETudoMais;  
significadoDaVidaDoUniversoETudoMais = 42;  
/* fim de código gerado por IA */
```

Exemplos de “modificações não significativas”: trocar nomes de variáveis e métodos, alterar formato ou estilo, modificar posições de linhas sem impacto na execução, substituir tipos de dados por seus equivalentes diretos, adicionar ou remover comentários ou código irrelevante, e outras coisas similares.

PRIMEIRA TAREFA:

Criar a classe Documento e sua hierarquia de classes que representa adequadamente as características dos diferentes tipos de documentos. Essa hierarquia está representada no diagrama abaixo similar a UML.

OBS.: quadrados vermelhos vazados indicam visibilidade privada e círculos verdes indicam visibilidade pública.



Os construtores das classes devem ser completos. A ordem dos parâmetros é a mesma ordem em que aparecem no diagrama UML, sendo que atributos de superclasses devem aparecer antes dos atributos da própria classe na sequência de parâmetros.

Os métodos de acesso (*getters*) devem estar disponíveis para todos os atributos e métodos de modificação (*setters*) são opcionais, já que um documento não muda depois de criado.

Você deve incluir comentários para documentação JavaDoc para todas as classes e todos os métodos que criar.

Essa primeira tarefa é avaliada pelo professor conforme a rubrica do final desse documento.

SEGUNDA TAREFA

Implementar o algoritmo para o burocrata despachar documentos com a maior eficiência possível, evitando aumentar seu estresse por descumprir regras administrativas ou por perder documentos de processos.

Novamente, você deve incluir comentários para documentação JavaDoc em todos os métodos do burocrata e nas classes auxiliares que eventualmente crie para ajudar a resolver o problema.

Duas coisas são críticas para a avaliação de eficiência: a quantidade total de processos transportados e evitar desperdício de espaço nos processos. Assim, despachar 4 documentos de 55 páginas em apenas um processo é mais eficiente do que despachar os mesmos 4 documentos em 4 processos diferentes.

Aumentar o estresse é pior do que ter uma eficiência baixa, pois prezamos pela qualidade de vida das pessoas no serviço público.

A documentação do programa (disponível em JavaDoc) também auxilia na identificação adequada dos elementos relevantes que devem ser considerados no comportamento do burocrata.

Essa segunda tarefa é usada de duas formas: a primeira é o teste do código do Burocrata pelo professor em várias condições de configuração da Universidade, sendo que erros e estresse não devem acontecer nesses testes; a segunda é a execução do código desenvolvido por estudantes em uma competição para conferir quem despacha documentos de forma mais eficiente.

Os testes do professor que avaliam erros de lógica fazem parte da avaliação do trabalho conforme a rubrica do final desse documento.

O resultado da competição NÃO faz parte da nota do trabalho, mas garantirá prêmios ou vantagens na componente curricular. A configuração exata do simulador que será usada na competição será revelada apenas no dia da competição, mas a fórmula exata para calcular a eficiência será revelada nas próximas semanas.

OBSERVAÇÃO 1: esse trabalho pode ser feito individualmente ou em dupla.

OBSERVAÇÃO 2: o enunciado desse trabalho pode sofrer pequenas alterações ao longo dos próximos dias para promover pequenos ajustes que tornem a atividade mais adequada aos objetivos da componente curricular ou mais interessante para estudantes.

OBSERVAÇÃO 3: dúvidas devem ser esclarecidas diretamente com o professor. Não assuma que suas hipóteses sobre o problema e a solução são verdadeiras, pois isso pode afetar o resultado dos testes do professor.

OBSERVAÇÃO 4: pode ser necessário apresentar o trabalho ao professor em um horário extraclasse combinado previamente. Se você for escolhida ou escolhido para apresentar e acabar não apresentando, você não terá nota no trabalho 1.

RUBRICA DE AVALIAÇÃO

Verifique como seu trabalho será avaliado conforme os critérios estabelecidos abaixo.

Critérios	Expectativa de desempenho para a entrega da PRIMEIRA TAREFA		
[20 pontos]	100%	50%	0%
Hierarquia de documentos	A estrutura de herança é compatível com o diagrama.	Problemas de ocultação ou problemas de visibilidade.	Falta de atributos ou métodos, ou problemas sérios na estrutura.
[20 pontos]	100%	50%	0%
Implementação de métodos auxiliares	Os métodos são implementados em todas as classes conforme as propriedades e assinaturas.	Os métodos são implementados com pequenos erros.	Faltam métodos auxiliares em algumas classes, ou há erros graves nas propriedades desses métodos em algumas classes.
[5 pontos]	100%	50%	0%
JavaDoc	Todas as classes e todos os métodos possuem comentários para documentação JavaDoc.	Mais da metade das classes e dos métodos possui comentários para documentação JavaDoc.	Menos da metade das classes e dos métodos possui comentários para documentação JavaDoc.
[5 pontos]	100%	50%	0%
Comentários, convenções e clareza	O código segue convenções da linguagem, e é organizado e muito claro e/ou apresenta comentários suficientes para entendimento completo.	O código falha em seguir as principais convenções da linguagem, ou é pouco claro e apresenta poucos comentários (insuficientes para entendimento completo), ou o código é pouco organizado (problemas sérios de indentação).	O código é obscuro e não possui nenhum comentário, ou o código é extremamente desorganizado ou foi entregue em um único arquivo .java.

Critérios	Expectativa de desempenho para a entrega da SEGUNDA TAREFA		
[20 pontos]	100%	50%	0%
Funcionamento geral	A simulação não tem erros e o burocrata executa ações.	A simulação apresenta erros intermitentes quando executa para algumas configurações específicas.	A simulação não funciona por erros ou inação do burocrata para qualquer configuração da universidade.
[20 pontos]	100%	50%	0%
Testes unitários	Todos os testes do professor são executados sem erros.	Até 50% dos testes do professor falham.	Mais de 50% dos testes do professor falham.
[5 pontos]	100%	50%	0%
JavaDoc	Todos os métodos de Burocrata e classes associadas possuem comentários para documentação JavaDoc.	Mais da metade dos métodos de Burocrata possui comentários para documentação JavaDoc.	Menos da metade dos métodos de Burocrata possui comentários para documentação JavaDoc.
[5 pontos]	100%	50%	0%
Comentários, convenções e clareza	O código segue convenções da linguagem, e é organizado, e muito claro e/ou apresenta comentários suficientes para entendimento completo.	O código falha em seguir as principais convenções da linguagem, ou é pouco claro e apresenta poucos comentários (insuficientes para entendimento completo), ou o código é pouco organizado (problemas sérios de indentação).	O código é obscuro e não possui nenhum comentário, ou o código é extremamente desorganizado ou foi entregue em um único arquivo .java.