

GAC111 – Técnicas de Programação Aplicada à Engenharia

Trabalho Prático

Objetivo do Trabalho

O principal objetivo deste trabalho é praticar os conceitos aprendidos na disciplina. Espera-se com isso que os alunos possam revisar os conteúdos vistos ao longo do curso e entendê-los melhor.

Um segundo objetivo é manter os alunos praticando a programação orientada a objetos, pois o aprendizado nesta disciplina só acontece ao se colocar em prática os conceitos apresentados nas aulas.

Proposta do Trabalho

O trabalho a ser desenvolvido corresponde a um **simulador implementado na linguagem Java**. Ele deve ser inspirado nos estudos de caso apresentados e discutidos nos Capítulos 10 (pg. 265) e 14 (pg. 394) do livro *Programação Orientada a Objetos com Java – Uma Introdução Prática Usando o BlueJ*, 4ª edição, dos autores Barnes e Kolling). Esse livro pode ser acessado a partir da biblioteca virtual da UFLA: http://www.biblioteca.ufla.br/pergamum/biblioteca_s/php/login_usu.php?flag=pearson_redirect.php.

Essencialmente, o simulador a ser implementado envolve o monitoramento de clientes dentro de uma área demarcada. Esse tipo de simulação é frequentemente utilizado para criar um modelo que permita avaliar o dimensionamento de atendentes para o atendimento das demandas dos clientes. Para que você tenha uma ideia do que se espera como produto final deste trabalho, como exemplo, está disponível no Campus Virtual um vídeo (videoExemplo.mp4) de um simulador de pedágio, onde os clientes são os veículos e os atendentes são os funcionários que ficam nas cabines de pedágio.

Nessa simulação, o objetivo principal é fazer com que clientes (por exemplo, veículos ou pessoas) se desloquem entre pontos distintos dentro de uma área demarcada para serem atendidos por algum atendente (por exemplo, uma pessoa ou uma máquina). Ao chegar até o atendente, o cliente demora um tempo para ser atendido e, depois, sai da simulação.

A simulação deve manter controle do tempo atual de simulação, independentemente da unidade de medida de tempo utilizada pelo sistema sendo modelado. Nessa simulação, a cada incremento do tempo (relógio), novos clientes podem chegar para atendimento, clientes podem se mover nas filas de espera por atendimento e clientes podem ter o atendimento finalizado.

Variáveis aleatórias devem ser utilizadas para modelar as componentes estocásticas do sistema, tais como a quantidade de clientes que chegam para ser atendidos em cada instante de tempo e a duração do atendimento de cada cliente. O valor máximo dessas variáveis aleatórias, a quantidade de atendentes e a duração da simulação devem ser parametrizáveis. Além disso, algum critério deve ser implementado para definir a distribuição dos clientes entre os atendentes.

Este trabalho **deverá ser feito utilizando o código inicial disponibilizado pelo professor**. O propósito deste trabalho é **melhorar e expandir** essa implementação inicial do simulador visando um bom projeto de classes e a adequada utilização de outros conceitos abordados na disciplina, tais como: herança, polimorfismo, métodos e classes abstratas, interfaces, coleções etc.

Cada grupo implementará uma proposta diferente de expansão desse simulador. A proposta a ser implementada por cada grupo será divulgada pelo professor assim que a formação dos grupos estiver concluída. Além disso, cada grupo deverá definir o objetivo da sua simulação.

Requisitos Adicionais

- O trabalho deverá usar corretamente os conceitos de Orientação a Objetos.
- O trabalho deverá fazer uso correto de composição e/ou agregação.
- O trabalho deverá usar corretamente pelo menos uma das estruturas de dados (lista, pilha, fila e mapa) estudada na disciplina.
- O trabalho deverá fazer uso correto de herança.
- O trabalho deverá fazer uso correto de polimorfismo (variável polimórfica e polimorfismo de método).
- O trabalho poderá fazer uso de padrões de projeto.
- O trabalho deverá ter um bom *Design* de Classes (baixo acoplamento, alta coesão, não ter replicação de código etc).
- O trabalho deve usar um pacote no padrão br.ufla.gac111.grupoX (onde X é o número do grupo). Obs.: se quiser, o grupo pode fazer uso de subpacotes.
- O trabalho deverá usar a GUI disponível no código fornecido pelo professor.
- Todo o código deve estar comentado (padrão Javadoc), incluindo os nomes dos autores de cada classe implementada no sistema.
- Também serão avaliados a legibilidade do código, organização e uso de comentários.
- Deve ser entregue junto com o trabalho o Diagrama de Classes UML da modelagem do sistema.
- Os requisitos aqui solicitados são requisitos mínimos, ou seja, o grupo pode acrescentar itens adicionais desde que não firam as regras estabelecidas nesta proposta.
- Não usar inferência de tipo nem expressão lambda na implementação do trabalho.

Pontuação e Entrega

Conforme previsto no Plano de Ensino, este trabalho vale 24% da nota do curso. Cada grupo deverá entregar um documento (*PropostaInicialTrabalhoPratico_Grupo@.pdf* – onde @ é o número do grupo no Campus Virtual) que definirá a proposta do sistema a ser implementado com o diagrama de classes UML. Esse documento (**PDF**) deve ser entregue via Campus Virtual **até 23h50 do dia 07/06/2023**.

Para a entrega final do trabalho deve haver uma pasta raiz contendo todo o código do trabalho (**somente arquivos .java**) e um arquivo **PDF** com o diagrama de classes UML. Essa pasta deve ter o nome *TrabalhoPratico_Grupo@*, onde @ corresponde à identificação (número) do grupo no Campus Virtual. A pasta deve ser **compactada (.zip) em um único arquivo** com o mesmo nome da pasta (ex: *TrabalhoPratico_Grupo@.zip*) e esse arquivo deve ser enviado pelo Campus Virtual **até 23h50 do dia 13/07/2023**.

Os alunos poderão utilizar qualquer editor e/ou IDE para o desenvolvimento do trabalho prático e não podem utilizar qualquer biblioteca ou API além do padrão Java (versão 8 ou superior). Além disso, **a implementação deverá compilar e executar corretamente a partir de um terminal**.

Todos os alunos devem participar ativamente da implementação do código do trabalho. Ainda que o trabalho seja desenvolvido em grupo, a avaliação será individual e levará em consideração a entrevista de cada aluno e a participação do mesmo no desenvolvimento do código.

ATENÇÃO: O grupo que não entregar o trabalho prático pelo Campus Virtual (a proposta inicial com diagrama de classes até 07/06/2023 e a implementação com diagrama de classes até 13/07/2023) ficará com nota zero e não participará da etapa de entrevista.