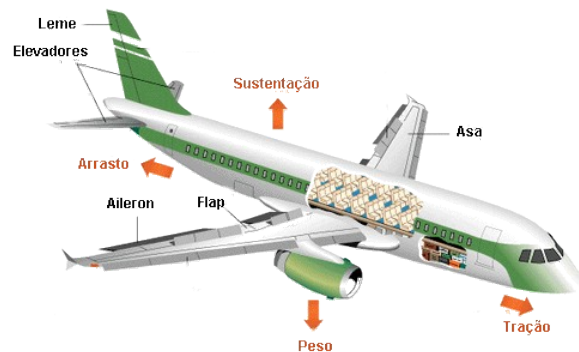


Aluno: _____

3ª Avaliação Individual – 2011.2

Questão 1) (2,0) Você foi contratado pela empresa *Tabajara Avionics Inc.* para desenvolver o sistema embarcado de controle das aeronaves de caça da empresa. O sistema é composto por uma série de sensores (altitude, velocidade, acelerômetro, temperatura, etc) e um grupo de atuadores (turbinas, *flaps*, *aileron*s, leme, trem de pouso etc). O sistema de controle faz com que um atuador (por exemplo o *flap*) dependa de informações obtidas de vários sensores, para que a aeronave funcione conforme o esperado. Alguns atuadores (por exemplo, o sistema de ejeção da cadeira do piloto) só são disparados após a ocorrência de uma sequência específica de eventos (por exemplo, aceleração alta + turbina em baixa rotação + alta temperatura). Projete o diagrama de classes para esta solução e indique quais padrões foram utilizados.



Questão 2) (2,0) O IFBa utiliza dois tipos diferentes de estrutura de dados para manter os registros dos seus alunos: lista encadeada e árvore. A lista encadeada contém os alunos ordenados pelo número de matrícula enquanto a árvore armazena os alunos agrupados por campus, departamento, curso e semestre de entrada (árvore de seis níveis incluindo as folhas). A depender do tipo de seleção sendo realizada a estrutura mais apropriada é utilizada. Diversos critérios de seleção poderão ser definidos e não são conhecidos *a priori*. Exemplos de seleção: alunos que nunca perderam disciplina, alunos de ADS que já cursaram mais que X disciplinas, etc. O cliente deverá executar uma determinada seleção S em cima de uma determinada estrutura E e gerar um relatório com os resultados. O cliente não deverá ser modificado para se adaptar a uma estrutura de dados ou seleção específica. Apresente o projeto de classes e os padrões de projeto utilizados.

Questão 3) (2,0) Você foi contratado pelo *Mercado Livre* para implementar o sistema de controle de compras *on-line*. Cada compra passa por uma série de passos até ser concluída, porém cada categoria de produto demanda um conjunto diferente de passos. Por exemplo, para livros o *workflow* básico é: *verificando estoque* → *confirmando pagamento* → *embalando* → *enviado*. Para computadores personalizados o *workflow* é: *confirmando aparelho* → *montando aparelho* → *realizando testes* → *embalando* → *enviado*. Cada compra pode ser: *modificada* ou *cancelada*, porém o comportamento destas operações depende do passo no qual a compra se encontra. A modificação ou cancelamento **não** depende da categoria do produto. Apresente o diagrama de classes da solução e os padrões de projeto utilizados.

Questão 4) (2,0) Você está desenvolvendo um *framework* para processamento de vídeos de sistemas de alta segurança. Um dos *hot-spots* do *framework* permite que o programador defina novas operações a serem realizadas no vídeo (efeitos, adição de comentários, etc). Para cada operação o *framework* deverá gerar um *log* indicando o usuário que a realizou e a duração da operação), sem entretanto exigir que o usuário do *framework* se preocupe com isso. Uma das operações a ser implementada é a criação de marcas (*snapshots*) em determinados momentos do vídeo. O número total de marcas é ilimitado e o vídeo poderá ser retrocedido para qualquer uma das marcas previamente criadas. Apresente o diagrama de classes e os padrões de projeto utilizados na sua solução.

Questão 5) (2,0) O *Visitor* é um padrão de projeto que permite a inclusão facilitada de novas operações a serem realizadas em um agregado heterogêneo de objetos. Adicionar novas operações específicas para cada tipo particular de elemento do agregado é fácil, porém adicionar um novo tipo de elemento do agregado requer a modificação da interface abstrata do *Visitor* e, conseqüentemente, de todos os *Visitors* concreto. Proponha uma solução para remediar este problema, suportando facilmente extensões nos dois *concerns* (operações e tipos de elementos), apresente o diagrama de classes e discuta as técnicas e eventuais padrões de projeto utilizados.

Boa sorte !