

Lista de Exercícios

MC302 - Programação Orientada a Objetos
Instituto de Computação
Universidade Estadual de Campinas

Classes
André Santanchè
2011

Questão 1

Escreva uma classe cujos objetos representam alunos matriculados em uma disciplina. Cada objeto dessa classe deve guardar os seguintes dados do aluno: matrícula, nome, 2 notas de prova e 1 nota de trabalho. Escreva os seguintes métodos para esta classe:

media	calcula a média final do aluno (cada prova tem peso 2,5 e o trabalho tem peso 2)
final	calcula quanto o aluno precisa para a prova final (retorna zero se ele não for para a final)

Questão 2

Escreva uma classe Data cuja instância (objeto) represente uma data. Esta classe deverá dispor dos seguintes métodos:

construtor	define a data que determinado objeto (através de parâmetro), este método verifica se a data está correta, caso não esteja a data é configurada como 01/01/0001
compara	recebe como parâmetro um outro objeto da Classe data, compare com a data corrente e retorne: <ul style="list-style-type: none">▫ 0 se as datas forem iguais;▫ 1 se a data corrente for maior que a do parâmetro;▫ -1 se a data do parâmetro for maior que a corrente.
getDia	retorna o dia da data
getMes	retorna o mês da data
getMesExtenso	retorna o mês da data corrente por extenso
getAno	retorna o ano da data
isBissexto	retorna verdadeiro se o ano da data corrente for bissexto e falso caso contrário
clone	o objeto clona a si próprio, para isto, ele cria um novo objeto da classe Data com os mesmos valores de atributos e retorna sua referência pelo método

Questão 3

Escreva uma classe em que cada objeto representa um voo que acontece em determinada data e em determinado horário. Cada voo possui no máximo 100 passageiros, e a classe permite controlar a ocupação das vagas. A classe deve ter os seguintes métodos:

construtor	configura os dados do voo (recebidos como parâmetro): número do voo, data (para armazenar a data utilize um objeto da classe Data, criada na questão anterior);
proximoLivre	retorna o número da próxima cadeira livre
verifica	verifica se o número da cadeira recebido como parâmetro está ocupada
ocupa	ocupa determinada cadeira do voo, cujo número é recebido como parâmetro, e retorna verdadeiro se a cadeira ainda não estiver ocupada (operação foi bem sucedida) e falso caso contrário
vagas	retorna o número de cadeiras vagas disponíveis (não ocupadas) no voo
getVoo	retorna o número do voo

getData	retorna a data do vôo (na forma de objeto)
clone	o objeto clona a si próprio, para isto, ele cria um novo objeto da mesma classe e faz uma cópia dos valores de seus atributos

Questão 4

Considere a seguinte classe, cujo método `respostaQuestao` recebe como parâmetro o número de uma questão e retorna a sua resposta correta, proveniente de um gabarito.

```
public class Gabarito
{
    public char respostaQuestao(int numeroQuestao)
    {
        :
    }
}
```

Escreva uma classe classe Prova em que cada objeto representa uma prova feita por um aluno. Esta prova possui 15 questões de múltipla escolha (letras A a E). As 10 primeiras questões valem 0,5 ponto e as 5 últimas questões valem 1 ponto. Esta classe deverá controlar as questões respondidas pelo aluno. Para isto, a classe deve implementar os métodos:

construtor	recebe como parâmetro um objeto da classe Gabarito contendo o gabarito da prova
respostaAluno	recebe como parâmetro a resposta dada pelo aluno a uma questão; este método não recebe entre os parâmetros o número da questão, ele mesmo deve estabelecer um controle interno de modo que as questões sejam inseridas sequencialmente, ou seja, a primeira vez que o método é chamado, insere a primeira questão, a segunda, insere a segunda questão, e assim por diante.
acertos	retorna a quantidade de questões que o aluno acertou
nota	retorna a nota que o aluno tirou na prova
maior	recebe como parâmetro um outro objeto da classe Prova e retorna a nota do aluno que acertou mais questões; se houver empate, retorna a maior nota; se houver empate novamente, retorna a nota do aluno representado no objeto corrente

Questão 5

A linguagem Java dispõe de um suporte nativo a vetores, que exige a definição de seu tamanho no momento da instanciação. Depois de instanciado, o tamanho do vetor não pode ser modificado. Escreva uma classe chamada Vetor cujos objetos simulem vetores de tamanho variável. A classe define os seguintes métodos:

construtor	recebe como parâmetro o tamanho inicial do vetor
insert	recebe como parâmetro uma string e a coloca na próxima posição disponível do vetor; note que o vetor cresce automaticamente, portanto, se a inserção ultrapassar o tamanho inicial estabelecido na criação, por exemplo, o vetor deve aumentar seu tamanho automaticamente
get	recebe como parâmetro uma posição do vetor e retorna a string que estiver naquela posição; se a posição não estiver ocupada ou ultrapassar o tamanho do vetor, este método retorna nulo
size	retorna o número de elementos inseridos no vetor (independente do tamanho do mesmo)

O java dispõe de classes - tal como a Vector - que realizam a tarefa solicitada nesta questão. Tais classes não devem ser usadas. É possível resolver esta questão apenas usando o sistema de vetores preexistente do Java, sem nenhum comando especial extra.

Questão 6

Cria uma classe VetorOrdenado que é uma variante da classe anterior, com as seguintes modificações e acréscimos:

insert	recebe uma string como parâmetro e a insere em uma posição tal que o vetor de nomes se mantenha sempre em ordem alfabética
merge	recebe como parâmetro um outro objeto da classe VetorOrdenado e retorna um terceiro objeto da classe VetorOrdenado (criado dentro do método) que contém o merge do vetor corrente com aquele recebido como parâmetro