

Aula 01 – Exercícios de Abstração

1) Os alunos da disciplina de Programação Orientada a Objetos da FCT/Unesp desenvolverão dois trabalhos e duas provas ao longo do semestre. Considere que, nesse caso, a média final é calculada pela média aritmética simples de todas as notas e que o aluno é aprovado somente se obtiver média maior ou igual a cinco.

O Aluno possui como informação o RA, Nome e as notas – a média é calculada a partir das notas e não precisa ser armazenada.

Aluno
nome:String; RA:String; t1:float; t2:float; p1:float; p2:float;
float calcularMedia();

2) Uma academia mantém registro de seus alunos armazenando nome, idade, peso e altura. A academia faz um desconto especial para menores de idade, portanto, é necessário saber distinguir entre um aluno maior e menor. Além disso, a academia também tem interesse em acompanhar o desempenho de seus alunos, por isso, ela também necessita conhecer o índice de massa corporal (IMC) deles, sendo que o $IMC = peso / altura ^ 2$.

Aluno
nome:String; idade:int; peso:float; altura:float;
boolean isMenor(); float IMC();

3) Um banco mantém contas de clientes armazenando o número da conta, o nome do cliente e o saldo atual da conta. Os clientes podem depositar o quanto quiserem nas contas, mas podem sacar somente enquanto a conta possuir saldo positivo.

Conta
numero:String; cliente:String; saldo:double;
depositar(valor:double); sacar(valor:double);

4) A Lista é uma estrutura de dados que pode ser representada computacionalmente de diferentes maneiras, por exemplo, um vetor, uma lista encadeada dinâmica, uma árvore, um *hash*, entre outras maneiras. A Lista, genericamente, pode ser vista como um tipo abstrato de dados em que as operações são conhecidas e comuns a qualquer tipo de implementação ou representação computacional. Faça a modelagem Orientada a Objetos

do tipo abstrato de dados de uma Lista, a qual possui elementos armazenados e as seguintes operações: inicializar lista, verificar se lista está vazia, verificar se a lista está cheia, adicionar um elemento, recuperar posição de um elemento, remover um elemento. Considere uma lista estática em vetor.

Lista
vetor[:int]; cont:int;
inicializar(); boolean estaVazia(); add(elemento:int); int buscar(element:int); remover(elemento:int);

5) A FCT/Unesp é organizada por vários departamentos de ensino e cada um deles é composto por docentes. Um docente só pode pertencer a um único departamento. Desenvolva uma modelagem para ilustrar a organização das classes para um programa orientado a objetos que contemple a estrutura organizacional da FCT/Unesp.

Da universidade é necessário armazenar somente o nome e os departamentos.

Os dados que deverão ser armazenados para o departamento são: Código, Nome, Telefone, Funcionários.

Os dados que deverão ser armazenados para o funcionário são: Código, Nome, Função, Salário

Universidade
nome:String; departamento[:Departamento];
addDepto(dep:Departamento);

Departamento
cod:String; nome:String; fone:String; funcionarios[:Funcionario];
add(func:Funcionario);

Funcionario
cod:String; nome:String; funcao:String; salario:double;