

CENTRO UNIVERSITÁRIO DE ANÁPOLIS
UNIEVANGÉLICA
CURSOS SUPERIORES DE COMPUTAÇÃO

DISCIPLINA: Programação Orientada a Objeto	CURSO: Engenharia da Computação
PROFESSOR: Alexandre Moraes Tannus	DATA:
ACADÊMICO (A): _____	

1. Elaborar um programa que entre nome, sala e quatro notas escolares de um aluno. O programa deve, ao final, apresentar o resultado da média do aluno. Para realizar essa tarefa, o programa deve usar uma classe que tenha nome, notas e média como campos públicos e sala como campo protegido, o qual necessitará de dois métodos de acesso, um para escrever no campo e outro para ler o campo.
2. Elaborar um programa que, a partir de uma classe chamada *Pessoa*, com os campos públicos Nome, Idade e Sexo, estabeleça um objeto X para dez elementos. O programa deve, por meio de um menu, dar a possibilidade de entrar todos os elementos, de apresentar todos os elementos e de sair do programa.
3. Elaborar um programa que tenha uma classe chamada *ContaCorrente* com os campos Número e Saldo e com os métodos Depositar e Sacar. O trecho de código do programa deve começar com o Saldo com valor zero e apresentar um menu com as opções depositar, sacar e finalizar, além de apresentar o valor do Saldo atual da conta. À medida que forem selecionadas as opções de ação do programa em depositar e sacar, o saldo deve ser apresentado sempre de forma atualizada.
4. Faça uma classe que simule o funcionamento de uma bomba d'água. A bomba possui um atributo booleano chamado "status" e os métodos "ligar" e "desligar" (ambos sem retorno). O método "ligar" coloca true em "status" e o método "desligar" coloca false em "status". A bomba deve ficar ligada durante um certo intervalo de tempo (em segundos). O tempo em segundos deve ser recebido pelo método ligar. A cada segundo, apresente em tela quantos segundos faltam para a bomba ser desligada. Decorrido o tempo, o método desligar é acionado e a bomba é desligada.
5. Crie uma classe chamada GPS contendo os seguintes atributos do tipo String: "idioma" e "rota". Defina dois métodos construtores: o default e outro para ligar o GPS com o idioma português e uma rota qualquer. Elabore métodos para realizar as seguintes funções:
 - a. Definir idioma
 - b. Definir rota
 - c. Um método chamado "mostrar" para apresentar todos os valores atuais dos atributos do GPS.Elabore também uma outra classe (UsaGPS) para testar essas funcionalidade

6. Considere a especificação a seguir:
7. “Um programa tem como finalidade funcionar como o controlador de um sistema de alarme contra ladrões. Um sistema típico é composto de vários sensores conectados por circuitos individuais a uma caixa de controle central contendo o controlador. A caixa de controle tem um teclado e uma tela simples. Os sensores incluem comutadores, detectores de calor e detectores de movimento. Cada sensor tem um código de identificação que pode ser lido pelo controlador para identificar o sensor.
8. O controlador permite que um operador selecione quais sensores estarão ativos e ligue ou desligue o sistema. Se um sensor for acionado quando o sistema estiver ativo o controlador deve ativar os alarmes (uma sirene e uma campainha) e exibir uma mensagem no painel da tela indicando qual sensor está envolvido. O operador deve inserir um código de segurança antes de o sistema ser ligado ou desligado.”
 - a. Identifique um conjunto de classes que possa ser usado na modelagem do sistema a partir de um ponto de vista orientado a objetos.
 - b. Use as classes para construir um diagrama de classes e exibir a estrutura do sistema.
 - c. Escreva um programa para simular o sistema, usando as classes.
9. Crie uma classe *Televisao* e uma classe *ControleRemoto* que pode controlar o volume e trocar os canais da televisão. O controle de volume permite:
 - a. aumentar ou diminuir a potência do volume de som em uma unidade de cada vez;
 - b. aumentar e diminuir o número do canal em uma unidade
 - c. trocar para um canal indicado;
 - d. consultar o valor do volume de som e o canal selecionado.
10. Escreva em Java a classe *NumeroComplexo* que represente um número complexo. A classe deverá ter os seguintes métodos:
 - *inicializaNumero*, que recebe dois valores como argumentos para inicializar os campos da classe (parte real e imaginária);
 - *imprimeNumero*, que deve imprimir o número complexo encapsulado usando a notação $a + bi$ onde a é a parte real e b a imaginária;
 - *éIgual*, que recebe outra instância da classe *NumeroComplexo* e retorna true se os valores dos campos encapsulados forem iguais aos da instância passada como argumento;
 - *soma*, que recebe outra instância da classe *NumeroComplexo* e soma este número complexo com o encapsulado usando a fórmula

$$(a + bi) + (c + di) = (a + c) + (b + d)i$$
 - *subtrai*, que recebe outra instância da classe *NumeroComplexo* e subtrai o argumento do número complexo encapsulado usando a fórmula

$$(a + bi) - (c + di) = (a - c) + (b - d)i;$$
 - *multiplica*, que recebe outra instância da classe *NumeroComplexo* e multiplica este número complexo com o encapsulado usando a fórmula

$$(a + bi) * (c + di) = (ac - bd) + (ad + bc)i;$$
 - *divide*, que recebe outra instância da classe *NumeroComplexo* e divide o número encapsulado pelo passado como argumento usando a fórmula

$$\frac{(a + bi)}{(c + di)} = \frac{ac + bd}{c^2 + d^2} + \frac{bc - ad}{c^2 + d^2}i$$