



LISTA DE EXERCÍCIOS I

ORIENTAÇÃO A OBJETOS

- 1) Escreva em Java uma classe *Contador*, que encapsule um valor usado para contagem de itens ou eventos. A classe deve oferecer métodos que devem:
 - a) Zerar;
 - b) Incrementar;
 - c) Retornar o valor do contador.

- 2) Escreva em Java uma classe *Ponto2D* que represente um ponto no plano cartesiano. Além dos atributos por você identificados, a classe deve oferecer os seguintes membros:
 - a) Construtores sobrecarregados que permitam a inicialização do ponto:
 - i) Por default (sem parâmetros) na origem do espaço 2D;
 - ii) Num local indicado por dois parâmetros do tipo *double* (indicando o valor de abcissa e ordenada do ponto que está sendo criado);
 - iii) Em um local indicado por outro ponto.
 - b) Métodos de acesso (getter/setter) dos atributos do ponto;
 - c) Métodos sobrecarregados de movimentação do ponto com os mesmos parâmetros indicados para os construtores;
 - d) Método de comparação semântica do ponto (*equals*);
 - e) Método de representação do objeto como *String*;
 - f) Método que permita calcular a distância do ponto que recebe a mensagem, para outro ponto;
 - g) Método que permita a criação de um novo ponto no mesmo local do ponto que recebeu a mensagem (*clone*);

- 3) Escreva em Java uma classe *NumeroComplexo*, que representa um número complexo. A classe deve fornecer as seguintes operações:
 - a) Construtor com valores das partes inteira e fracionária;
 - b) Métodos *getter/setter* para os atributos da parte inteira e parte imaginária;
 - c) Método *somar*, que recebe outro número complexo e o adiciona ao número complexo que recebeu a mensagem. $(a+bi)+(c+di) = (a+c)+(b+d)i$;
 - d) Método *subtrair*, que recebe outro número complexo e o subtrai do número complexo que recebeu a mensagem. $(a+bi)-(c+di) = (a-c)+(b-d)i$;
 - e) Método *multiplicar*, que recebe outro número complexo e o multiplica ao complexo que recebeu a mensagem: $(a+bi) * (c+di) = (ac-bd)+(ad+bc)i$;
 - f) Método *dividir*, que recebe outro número complexo e o divide ao complexo que recebeu a mensagem: $(a+bi) / (c+di) = (ac+bd)/(c^2 + d^2) + (bc-ad)/(c^2 + d^2)i$;
 - g) Um método de comparação semântica dos números complexos;

- h) Um método que gere e retorne a representação string do número complexo;
 - i) Um método que retorne o módulo do número complexo.
- 4) Escreva em Java uma classe que represente uma reta ($y=ax+b$). Forneça os seguintes membros de classe:
- a) Construtores sobrecarregados que criem uma reta a partir de:
 - i) Dois valores, representando o coeficiente angular e o coeficiente linear da reta;
 - ii) Dois pontos;
 - b) Métodos de acesso para o coeficiente angular e para o coeficiente linear da reta;
 - c) Um método que verifique se um ponto dado pertence a reta;
 - d) Um método que gere e retorne a representação String da reta;
 - e) Um método que dada uma outra reta, retorne o ponto de interseção da reta dada ou null se as retas forem paralelas.
- 5) Escreva em Java uma classe que represente um círculo no plano cartesiano. Forneça os seguintes membros de classe:
- a) Um construtor que receba o raio e um ponto (o centro do círculo);
 - b) Um construtor que receba o raio e posicione o círculo na origem do espaço cartesiano;
 - c) Métodos de acesso ao atributo raio do círculo;
 - d) Métodos inflar e desinflar, que, respectivamente, aumentam e diminuem o raio do círculo de um dado valor;
 - e) Métodos sobrecarregados, inflar e desinflar, que, respectivamente, aumentam e diminuem o raio do círculo de uma unidade;
 - h) Métodos sobrecarregados mover, que:
 - i) por default (sem parâmetros) levam o círculo para a origem do espaço 2D;
 - ii) movem o círculo para um local indicado por dois parâmetros do tipo double (indicando o valor de abcissa e ordenada do ponto para onde o círculo se move);
 - iii) movem o círculo para o local indicado por outro ponto.
 - f) Método que retorna a área do círculo
- 6) Escreva uma classe que represente um país. Um país é representado através dos atributos: código ISO 3166-1 (ex.: BRA), nome (ex.: Brasil), população (ex.: 193.946.886) e a sua dimensão em Km² (ex.: 8.515.767,049). Além disso, cada país mantém uma lista de outros países com os quais ele faz fronteira. Escreva a classe em Java e forneça os seus membros a seguir:
- a) Construtor que inicialize o código ISO, o nome e a dimensão do país;
 - b) Métodos de acesso (getter/setter) para as propriedades código ISO, nome, população e dimensão do país;
 - c) Um método que permita verificar se dois objetos representam o mesmo país (igualdade semântica). Dois países são iguais se tiverem o mesmo código ISO;
 - d) Um método que informe se outro país é limítrofe do país que recebeu a mensagem;
 - e) Um método que retorne a densidade populacional do país;
 - f) Um método que receba um país como parâmetro e retorne a lista de vizinhos comuns aos dois países.
- Considere que um país tem no máximo 40 outros países com os quais ele faz fronteira.

- 7) Escreva em Java uma classe Continente. Um continente possui um nome e é composto por um conjunto de países. Forneça os membros de classe a seguir:
- a) Construtor que inicialize o nome do continente;
 - b) Um método que permita adicionar países aos continentes;
 - c) Um método que retorne a dimensão total do continente;
 - d) Um método que retorne a população total do continente;
 - e) Um método que retorne a densidade populacional do continente;
 - f) Um método que retorne o país com maior população no continente;
 - g) Um método que retorne o país com menor população no continente;
 - h) Um método que retorne o país de maior dimensão territorial no continente;
 - i) Um método que retorne o país de menor dimensão territorial no continente;
 - j) Um método que retorne a razão territorial do maior país em relação ao menor país.
- 8) Escreva uma classe Pessoa que representa uma pessoa numa árvore genealógica. A pessoa possui um nome, um pai e uma mãe (que também são pessoas). Forneça os seguintes membros para a classe:
- a) Construtores sobrecarregados que:
 - i) inicialize o nome da pessoa, bem como seus antecessores (pai e mãe);
 - ii) inicialize o nome da pessoa, e coloque seus antecessores para null;
 - b) Um método que verifique a igualdade semântica entre duas pessoas (as pessoas são iguais se possuem o mesmo nome e a mesma mãe);
 - c) Um método que verifique se duas pessoas são irmãs;
 - d) Um método que verifique se uma pessoa é antecessora da pessoa que recebeu a mensagem (é seu pai ou sua mãe, ou antecessor do pai ou antecessor da mãe).
- 9) Escreva uma classe Conjunto, que represente um conjunto de tamanho variável (crescimento de array por demanda) de elementos do tipo String. Escreva os seguintes membros para a classe:
- a) Um método que permita adicionar um elemento para o conjunto (o elemento não pode existir no conjunto);
 - b) Um método que permita verificar se um dado elemento pertence ao Conjunto;
 - c) Um método uniao, que retorne um novo conjunto de acordo com a semântica da operação união entre conjuntos (um novo conjunto, sem elementos repetidos, com a combinação dos elementos dos dois conjuntos originais, o que recebeu a mensagem e o que foi passado como parâmetro);
 - d) Um método inter, que retorne um novo conjunto de acordo com a semântica da operação interseção entre conjuntos (um novo conjunto, sem elementos repetidos, com os elementos que estejam nos dois conjuntos originais, o que recebeu a mensagem e o que foi passado como parâmetro);
 - e) Um método menos, que retorne um novo conjunto de acordo com a semântica da operação subtração entre conjuntos (um novo conjunto, sem elementos repetidos, com os elementos do conjunto que recebeu a mensagem, e que não existam no conjunto passado como parâmetro).

- 10) Crie uma classe Matriz que represente uma matriz matemática. Forneça um construtor que permita a inicialização das dimensões da Matriz
- Forneça métodos para acesso (leitura/escrita) de cada elemento da matriz.
- Forneça os métodos adequados para as seguintes operações com matriz:
- a) Comparação semântica da matriz;
 - b) Retornar a transposta (é aquela onde as linhas se transformam em colunas e as colunas em linhas) da matriz.
 - c) Retornar a oposta (é aquela onde todos os elementos possuem sinais trocados) da matriz;
 - d) Gere uma matriz nula (é aqueles onde todos os elementos são iguais a 0);
 - e) Informe se a matriz é identidade (matriz quadrada onde os elementos da diagonal principal são todos iguais a 1 e os demais 0);
 - f) Informe se a matriz é diagonal (matriz quadrada onde os elementos fora da diagonal principal são todos iguais a 0).
 - g) Informe se a matriz é singular (matriz diagonal onde os elementos da diagonal principal são todos iguais);
 - h) Informe se a matriz é simétrica (uma matriz quadrada é dita simétrica se ela é igual a sua transposta);
 - i) Informe se a matriz é anti-simétrica (uma matriz quadrada é dita anti-simétrica se sua oposta é igual a sua transposta)
 - j) Adicionar duas matrizes (alterando o valor da que recebeu a mensagem);
 - k) Subtrair duas matrizes(alterando o valor da que recebeu a mensagem);
 - l) Multiplicar duas matrizes(alterando o valor da que recebeu a mensagem);
 - m) Gere uma cópia da matriz.