ADS – IFPB – Campus Monteiro

LISTA SEMANAL – PROGRAMAÇÃO II - PROGRAMAÇÃO ORIENTADA A OBJETOS  
Prof. Cleyton Caetano de Souza

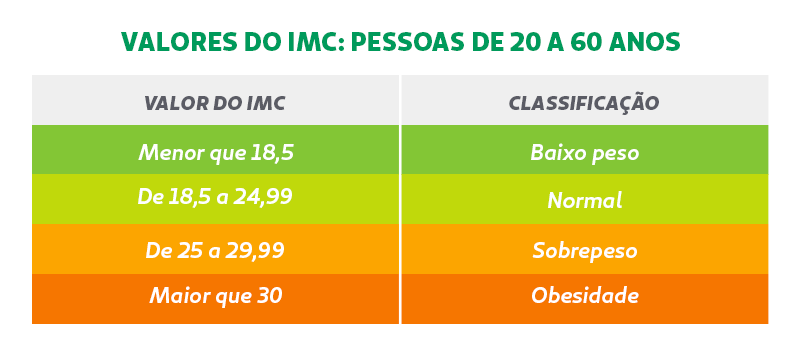
**Observação**: essa lista pode ser feita em dupla.

Semana 3

1. O operador ternário é uma alternativa mais resumida para substituir o “if-else”, em algumas situações específicas. **Pesquise sobre o funcionamento do operador ternário** e, considerando isso, preencha a tabela de acompanhamento com o valor das variáveis x, y, z e w, ao final de cada linha, durante a execução do programa abaixo:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **linha** | **x** | **y** | **z** | **w** | | **3** | 3 | 0 | 0 | 0 | | **4** |  |  |  |  | | **5** |  |  |  |  | | **6** |  |  |  |  | | **7** |  |  |  |  | |

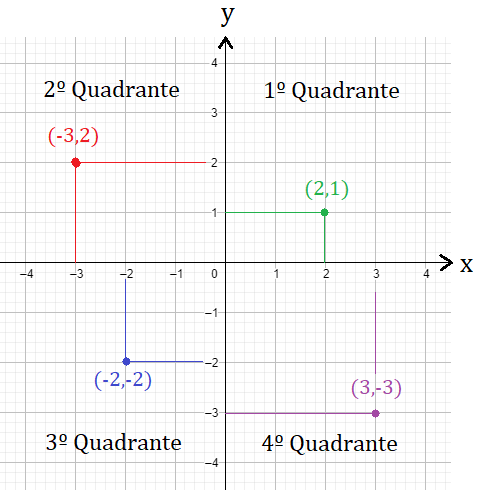
2. Crie uma classe chamada **Paciente**, com os atributos peso (float) e altura (float). Siga as convenções de nomenclatura e visibilidade vistas em sala de aula. Crie uma classe chamada **Nutricionista**, que será responsável por fazer a avaliação física do Paciente, através do cálculo do seu IMC. O IMC é o resultado da divisão do peso do paciente dividido pela altura elevada ao quadrado (). A classe **Nutricionista** possuirá um método chamado **avaliarIMC**, o qual receberá um objeto do tipo Paciente e retornará uma String correspondente à classificação do IMC do Paciente seguindo os seguintes critérios:



3. Escreva um programa em que você lerá os dados de um paciente e usará o nutricionista para exibir qual a classificação do paciente.

4. Sobre o método **avaliarIMC**, você acha que o design proposto para esse método ficou bom? Justifique a sua resposta.

5. Em Java, Enum é a criação de um tipo com valores pré-definidos. Pesquise sobre “Enum” e crie um Enum para representar as possíveis classificações do IMC (o nome do Enum pode ser ResultadoIMC e os valores do Enum são as categorias). Faça as alterações necessárias, no Nutricionista (o retorno do método avaliarIMC deve passar a ser um valor do tipo “ResultadoIMC”) e no programa, para poder fazer uso do Enum.

6. Crie uma classe chamada Ponto. Na matemática, um ponto é uma representação no plano cartesiano do encontro de duas coordenadas, uma no eixo horizontal e outra no eixo vertical. Por isso, a classe ponto vai possuir dois atributos, as coordenadas e , do tipo inteiro. Respeite as convenções de nomenclatura e visibilidade vistas em sala de aula (crie gets e sets para esses atributos).

7. Adicione à classe Ponto um método chamado eIgual (lê-se “é igual”). O método eIgual receberá como parâmetro de entrada outro objeto do tipo ponto e retornará um valor booleano indicando se o ponto recebido como parâmetro possui **o mesmo estado** – i.e., os mesmos valores das coordenadas e ) do objeto que foi usado para chamar o método. Para isso, basta comparar as propriedades do objeto recebido como parâmetro de entrada com as propriedades da própria classe.

8. Adicione à classe Ponto o método “quadrante” que não recebe parâmetro de entrada e retorna uma String, representando o quadrante ao qual o ponto pertence, conforme a figura ao lado. Se alguma das coordenadas do ponto for zero, o método retorna o eixo em que ele está (eixo horizontal ou eixo vertical). Se as duas coordenadas forem zero, o método retorna “origem”.

9. Crie um Programa para testar os métodos quadrante e eIgual, da classe Ponto. Comece perguntando ao usuário quais as coordenadas do primeiro ponto, então instancie um ponto e “sete” o valor dos seus atributos e informe em qual quadrante ele está localizado. Em seguida, pergunte ao usuário quais as coordenadas do segundo ponto, instancie outro ponto e “sete” o valor dos seus atributos e informe em qual quadrante ele está localizado. Por último, use o método eIgual de um dos objetos ponto para compará-lo com o outro ponto e exiba a mensagem “são iguais” se o retorno for true e “não são iguais” se o retorno for false.

|  |  |
| --- | --- |
| **Console I**  X 1 = 10  Y 1 = 10  O primeiro ponto está no 1º quadrante.  X 2 = 3  Y 2 = 5  O segundo ponto está no 1º quadrante.  Os dois pontos são diferentes. | **Console II**  X 1 = -10  Y 1 = -2  O primeiro ponto está no 3º quadrante.  X 2 = -10  Y 2 = -2  O segundo ponto está no 3º quadrante.  Os dois pontos são iguais. |

10. Escreva um programa onde você vai ler dois nomes fornecidos pelo usuário. Em seguida, indique se os nomes fornecidos são iguais ou não. **Lembre-se, não se deve comparar objetos usando o operador de igualdade, por isso pesquise sobre como comparar os nomes utilizando a classe String.**