**Instituto Nacional de Telecomunicações – INATEL**

**Laboratório de C206 – POO**

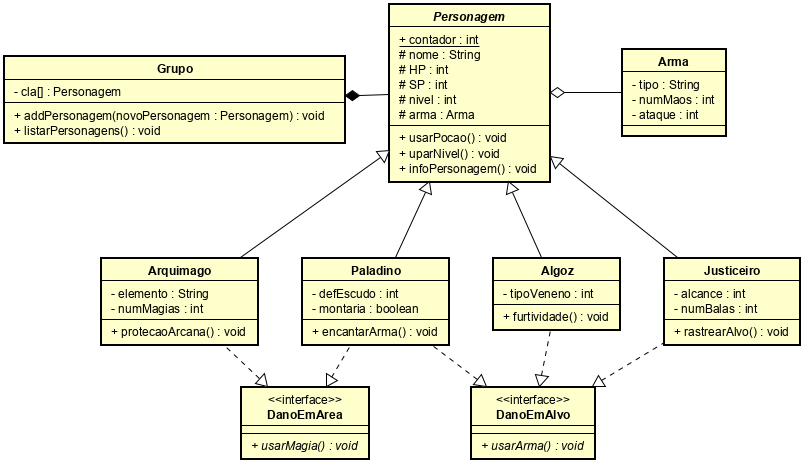
**Segunda Avaliação**

**Professor:** Renzo P. Mesquita

**Monitor:** Samuel da Cruz Souza

**Nome do aluno: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Turma: \_\_\_\_\_\_**

Ragnarök Online é um jogo de MMORPG baseado na série em quadrinhos (coreana) Ragnarök. No jogo, você controla um personagem que possui uma classe, que define qual a especialidade de seu personagem. No jogo, você pode escolher entre Arquimago, Paladino, Algoz e diversas outras classes. Também é possível participar de clãs, que são onde vários jogadores costumam se reunir. Sua tarefa é desenvolver um programa para fazer uma listagem de personagens.

1. Desenvolva um programa para organizar os diferentes tipos de classes, seguindo o diagrama abaixo:

Na classe principal:

Crie pelo menos um objeto de cada classe e salve dentro do array *exercito[]* através do método *addPersonagem*. Após salvar os personagens, execute o método *listarPersonagens* e mostre a quantidade de personagens no array através da variável estática *contador*.

**Atenção! Dentro do método *listarPersonagens* execute todos os métodos da classe que está sendo listada!**

**Notas importantes:**

* A classe Personagem é abstrata;
* Dentro dos métodos *usarMagia()* e *usarArma()* mostre uma mensagem indicando qual método está sendo usado e também o tipo de arma do personagem;
* No método *usarPocao()* incremente o HP em 50 unidades e no método *uparNivel()* incremente o nível em uma unidade;
* Não é necessário entrada de dados, preencha todos os atributos dentro da classe main;
* Preencha todos os atributos das classes, não se esqueça de criar todos os Getters e Setters;

**Dicas:**

* Ao reescrever o método *mostraInfo* nas classes filhas, chame o método já criador através do atributo **super.mostraInfo()** para poupar código;
* Instancie a classe Arma fora do construtor (dentro da classe Personagem);
* Para implementar mais de uma interface, usa-se a vírgula. Exemplo: ***implements*** *Interface1, Interface2*
* Ao salvar um personagem no array, use um **break** para evitar que todas as posições do array sejam preenchidas com o mesmo valor;
* Caso ache necessário, crie um método *mostrarInfo* dentro da classe Arma;