

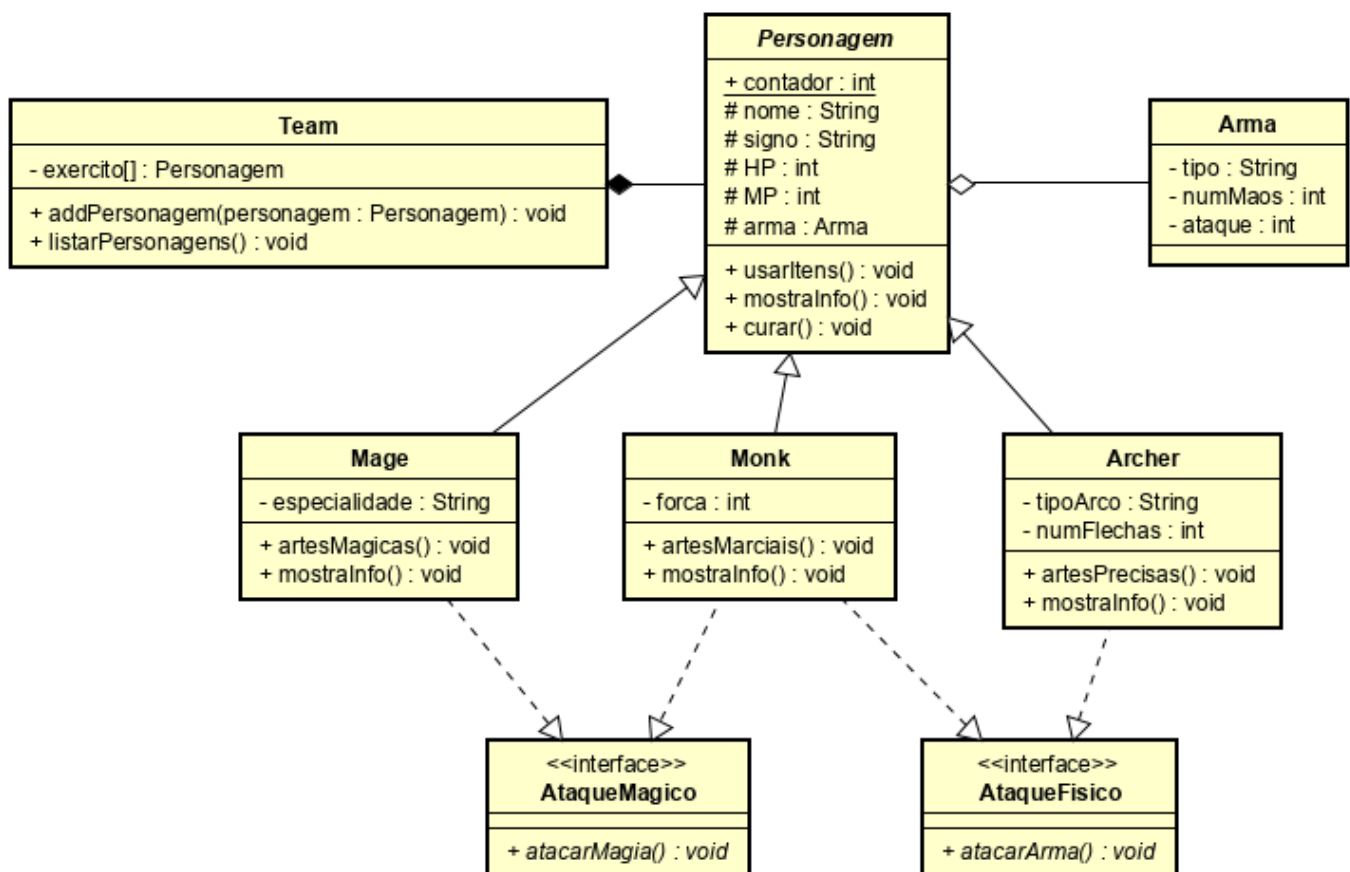
Instituto Nacional de Telecomunicações – INATEL
Laboratório de C206 – POO
Segunda Avaliação

Professor: Renzo P. Mesquita

Monitor: Samuel da Cruz Souza

Nome do aluno: _____ **Turma:** _____

Final Fantasy Tactics é um jogo de RPG tático desenvolvido e publicado pela SquareSoft para o PlayStation. O jogo consta com um sistema de classes, onde é possível que seu personagem tenha uma profissão, como por exemplo Mago, Cavaleiro ou até mesmo um Monge. Seu objetivo é organizar um exército onde serão inseridos todos os personagens, conforme o diagrama de classes:



1. Desenvolva um programa estruturado conforme o diagrama acima. Na classe principal (Main) crie no mínimo um personagem de cada tipo (mage, monk e archer), preencha todos os seus atributos e adicione ao seu exército através do método *addPersonagem*. Execute todos os métodos de cada personagem e ao final liste todos os personagens, e mostre quantos foram inseridos no array através da variável estática *contador*.

Notas Importantes:

- A classe Personagem deve ser abstrata e a variável contador do tipo estática;
- No método *atacarMagia()* mostre a especialidade da classe seguido de uma mensagem indicando que o método está sendo chamado;
- No método *atacarArma()* mostre o tipo de arma que seu personagem possui, seguido de uma mensagem indicando que o método está sendo chamado. Se for usado pela Classe Archer deverá consumir uma unidade de flecha;
- No método *curar()* faça com que o HP do personagem seja incrementado em 10 unidades;
- Não é necessário fazer entrada de dados, todos os atributos deverão ser preenchidos na classe main.
- Preencha todos os atributos das classes, não esqueça dos Getters e Setters para os atributos do tipo private!!

Dicas:

- Tente executar todos os métodos do personagem dentro do método *listarPersonagens*, execute por último o método *mostraInfo*;
- 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;