

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ - Campus de Curitiba (Brasil) -
Departamento Acadêmico de Eletrônica (DAELN). Disciplina: **Fundamentos de Programação 2 - IF62C**/Turma: **S11**. Prof: Jean M. Simão. Curso: **Engenharia Industrial Elétrica, ênfase Eletrônica/Telecomunicações**. Prova sobre linguagem C++ – Prova da 1ª Parcial.

Nome do Aluno: _____

Horário de Começo: _____

Horário de Fim: _____

Leia toda a prova antes de começar, pois os enunciados estão completados uns nos outros.

(Questão 1) Em um programa C++ (para *console*), além de uma classe *Principal_Jogo*, crie também uma classe abstrata chamada de *Personagem*. Esta classe terá (pelo menos) um atributo *id*.

Isto feito, crie uma classe *Jogador* e uma classe *Inimigo*, ambas derivadas de *Personagem* e cada qual com uma função-membro chamada “*void informe_papel()*”. No caso de *Jogador* tal função-membro informará (em tela) “*mocinho*”, enquanto que no caso de *Inimigo* informará “*bandido*”. Ainda, crie pelo menos um atributo pertinente e respectivos métodos para cada uma destas duas classes.

(Questão - 2)

A classe *Jogador* terá um atributo chamado *num_vidas* que deverá (nos seus objetos) ser acessível de alguma maneira, por algum método apropriado.

Ainda com relação à classe *Jogador*, o operador de fluxo (<<) deverá ser sobrecarregado para suportar objetos dela, de maneira que cada objeto de *Jogador* possa ter o valor de seus atributos (incluindo os derivados) impressos em tela.

(Questão - 3) Crie uma classe genérica ou gabarito chamada *Elemento*. Esta classe será útil para compor listas simplesmente encadeadas relativas a (ponteiros de) objetos de uma dada classe.

(Questão - 4) Crie uma classe *Lista_Personagens* para (ponteiros de) objetos de subclasses de *Personagem*. Cada objeto (potencial) da classe *Lista_Personagens* pode ter um número variável de elementos inclusos. Salienta-se ainda que cada objeto de subclasse de *Personagem* pode (potencialmente) participar de diversos objetos de *Lista_Personagens*. Assim sendo, utilize a classe *Elemento<Tipo>* (previamente criada) na implementação desta classe *Lista_Personagens*.

Obs.:

- (a) Utilizar alocação dinâmica de memória para implementar a lista encadeada.
- (b) A classe *Lista_Personagens* deverá ter um método para listar bem como outro para incluir.
- (c) A listagem apresentará o *id* de cada objeto e o seu ‘*papel*’ no jogo.

(Questão - 5)

- Faça com que a classe *Principal_Jogo* tenha (um atributo privado que será) um objeto da classe *Lista_Personagens* chamado *obj_Lista_P_1*.

- A classe *Principal_Jogo* deverá ter também um método para listar os objetos de subclasses de *Personagem*, bem como outro método para incluí-los. Estes métodos chamarão métodos do objeto *obj_Lista_P_1* passando os parâmetros necessários (quando necessários).

- A classe *Principal_Jogo* deverá ter ainda um método *menu* que, por meio de um laço de repetição, permitirá: (1) Criar Jogador; (2) Criar Inimigo; (3) Listar Personagens; ou (4) Sair.

Obs. gerais:

- (a) Utilizar *cout/cin* para entrada/saída, bem como *string* em vez de vetores de caractere ordinários.
- (b) Utilizar os bons princípios da orientação a objetos.

A interpretação faz parte do conteúdo da prova! Inclua comentários (se for o caso) para deixar claras suas decisões.