

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 2ª 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) No âmbito de um programa C++ (para *console*), crie uma classe abstrata chamada de *Personagem* com uma única função virtual-pura e de acordo com as demais questões propostas nesta prova. Ainda, esta classe terá (pelo menos) um atributo *id* que certamente não será público.

(Questão 2) Crie uma classe *Jogador* e uma classe *Inimigo*, ambas derivadas (publicamente) de *Personagem* e cada qual com uma função-membro chamada “*void informe_papel()*” que é a redefinição de uma dada função da classe base. No caso de *Jogador*, tal função-membro informará (em tela) “*Do Bem*”, enquanto que no caso de *Inimigo* informará “*Do Mal*”. Ainda, crie pelo menos um atributo pertinente do tipo *string* e respectivos métodos para cada uma destas duas classes.

(Questão - 3) 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 - 4) Crie uma classe *Lista_Personagens* para ponteiros de objetos de subclasses de *Personagem*. Esta classe terá como atributo principal uma instância de *List* (da *STL*) parametrizada com ponteiro de *Personagem*. Ainda, a classe *Lista_Personagens* deverá certamente ter um método para incluir bem como outro para listar. Por fim, a listagem apresentará o *id* de cada objeto e o seu ‘papel’ no jogo.

(Questão - 5) – Crie uma classe *Principal_Jogo* e faça com que 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 seu objeto: (1) criar Jogador(es), certamente via alocação de memória dinâmica e certamente atribuindo valores a seus atributos a partir de dados fornecidos pelo usuário; (2) criar Inimigo(s), certamente nos mesmos moldes da criação de Jogador(es); (3) Listar Personagens; ou (4) Sair.

- A classe *Principal_Jogo* deverá, por fim, fazer com que seu objeto libere memória alocada em método apropriado para tal que é chamado em momento oportuno.

Obs. Utilizar: (a) *cout/cin* para entrada/saída e (b) os bons princípios da orientação a objetos.