

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ - Campus de Curitiba
(Brasil) - Departamento Acadêmico de Eletrônica (DAELN). Disciplina: **Fundamentos
de Programação 2 - IF62C/Turma: S11/S12. Prof: Jean M. Simão.**

Curso: **Engenharia Eletrônica.**

Prova sobre linguagem Orientação a Objetos – Prova da 1ª Parcial.

Nome do Aluno: _____ Turma: _____

Horário de Começo: _____ Horário de Fim: _____

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

Utilize os bons princípios de projeto e programação orientada a objetos.

(Questão - 1) Em um programa C++ (para *console*), crie uma classe abstrata chamada de *Nave* com uma única função virtual pura chamada “*void informe_natureza()=0*” podendo, entretanto, ter outras funções que não sejam virtuais puras, incluindo construtora e destrutora. Esta classe terá um atributo protegido alfanumérico chamado *número*, certamente com seus respectivos métodos *get* e *set*. Ainda, quando pertinente, usar elementos constantes (i.e. atributos, funções, retornos e parâmetros constantes) nesta classe, assim como nas demais classes solicitadas nas demais questões.

(Questão - 2) Crie uma classe *Lista_Naves* (em C++) para ponteiros de objetos de subclasses de *Nave* baseando-se nos princípios de lista simplesmente encadeada. Ainda, a classe *Lista_Naves* deverá ter um método para incluir, um método para excluir e outro para listar. Particularmente, a cada chamada, o método incluir permitirá incluir (um ponteiro de objeto de subclasse de *Nave*) sempre na última posição da lista. Por sua vez, o método excluir permitirá excluir (um ponteiro) de qualquer posição da lista conforme parâmetro (de posição na lista) dado na chamada do método, sendo o valor *default* deste parâmetro a última posição. Por fim, a listagem apresentará a ‘natureza’ de cada objeto.

(Questão - 3) Em ++, crie uma classe *Avião* (com atributo privado booleano *supersônico* e seus respectivos *get* e *set*) e uma classe *Navio* (com um atributo privado inteiro *capacidade* e seus respectivos *get* e *set*), ambas derivadas de *Nave* e cada qual com uma função-membro chamada “*void informe_natureza()*”. No caso de *Avião* tal função-membro informará (em tela) “*Transporte Aéreo*”, enquanto que no caso de *Navio* informará “*Aquático*”. Por fim, sobrecarregue o operador de igualdade (*operator==*) na classe *Navio* de maneira que um objeto desta possa comparar todos seus atributos (inclusive o conjunto de atributos herdados) para com todos os atributos de outro objeto desta mesma classe.

(Questão - 4) Em C++, crie uma classe *Principal* cujo único objeto (ou instância) poderá armazenar em si ponteiros de objetos *Avião* por meio de uma função *void incluir_Avião (Avião* pa)*, bem como poderá armazenar em si ponteiros de objetos *Navio* por meio de uma função *void incluir_Navio (Navio* pn)*. Entretanto, estes apontamentos de objetos *Avião* e *Navio* serão armazenados como apontamentos de objetos *Nave*, por meio de *static_cast*, em uma lista para tal.

(Questão - 5) Elabore um Diagrama de Classes (em *UML*) que modele as classes requisitadas nas questões precedentes, bem como seus relacionamentos.