



INSTITUTO FED. DE EDUCAÇÃO, CIÊNC. E TEC. DE PERNAMBUCO
CURSO: TEC. EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
DISCIPLINA: ALGORITMOS E ESTRUTURAS DE DADOS
PROFESSOR: RAMIDE DANTAS
ASSUNTO: PILHAS, FILAS E LISTAS COM PONTEIROS

Aluno (a):			
Matrícula:		Data:	

Prática 05

OBS: Essa prática faz uso de conhecimentos exercitados na Prática 4.

Parte 0: Preparação

Passo 1: Crie um novo projeto chamado **Pratica5**.

Esta prática é uma continuação da prática 4 e faz uso dos mesmos códigos que acompanham essa prática.

Parte 1: Implementando Pilhas com Ponteiros

Passo 1: Utilize o arquivo **polonesa.cpp** que acompanha a prática 4 ao projeto.

Esse arquivo implementa uma calculadora polonesa simples baseada em pilha, conforme explicado em sala de aula.

Passo 2: Crie um arquivo chamado **pilha2.h** e implemente nele a classe `Pilha` usando ponteiros e estruturas encadeadas, como descrito no material de aula.

Modifique o arquivo **polonesa.cpp** para incluir **pilha2.h**. Também pode ser necessário fazer outras modificações menores

Passo 3: Compile e teste a aplicação, verificando se o resultado é o esperado.

Modifique o programa, adicionando novas expressões e verificando o resultado.

Parte 2: Implementando Filas com Ponteiros

Passo 1: Utilize o arquivo **impressora.cpp** da prática 4 ao projeto.

Assim como na prática 4, será preciso configurar o projeto para usar o padrão C++11 para compilar.

Passo 2: Crie o arquivo **fila2.h** e implemente a classe `Fila` utilizando ponteiros conforme descrito no material de aula.

Modifique **impressora.cpp** para incluir esse arquivo. Faça os ajustes necessários.

Passo 3: Compile e teste a aplicação, verificando se o resultado é o esperado.

Verifique se o comportamento está coerente com o observado na prática 4.

Parte 3: Implementando Listas com Ponteiros (Listas Encadeadas)

Passo 1: Adicione o arquivo **lista.cpp** da prática 4 ao seu projeto.

Passo 2: Crie o arquivo **lista2.h** e implemente a classe `Lista` usando ponteiros conforme descrito no material de aula.

Modifique **lista.cpp** de forma a incluir esse arquivo. Faça os ajustes que forem necessários

Passo 3: Compile e teste a aplicação, verificando se o resultado é o esperado.

Faça modificações nas adições, inserções e remoções para testar o comportamento da lista.

Parte 4: Refatore e reorganize o código

Passo 1: Refatore o código de forma que as duas implementações (com Array e Encadeada) da classe `Pilha` sejam subclasses de uma classe abstrata `Pilha`, que define a interface dessas estruturas.

Por exemplo, crie as classes `PilhaArray` e `PilhaEncadeada`, que descendem de `Pilha` genérica. Organize os arquivos `.h` de forma coerente.

Passo 2: Adeque o código de **polonesa.cpp** de forma que haja um ponteiro para `Pilha` genérica que pode ser instanciado tanto com `PilhaArray` ou `PilhaEncadeada`.

Passo 3: Repita os passos acima para as estruturas `Fila` e `Lista` e seus respectivos arquivos `cpp`.