

Trabalho Prático 01 - AEDS 1 - TRIO

Professora: Thais R. M. Braga Silva

Valor: 15 pontos

Data de Entrega (código + documentação): 11/10/2022

Data das Entrevistas: 17/10 (monitoria), 18/10 e 20/10 (laboratórios)

Forma de Entrega: PVANet (formato .zip ou .tar.gz)

Neste trabalho, vamos implementar listas lineares e TADs para gerenciar a ocorrência de palavras em um texto. A ideia é montar um TAD Dicionário contendo listas das palavras que o texto possui e que começam com cada letra do alfabeto. Dessa forma, serão até 26 listas, cada uma delas contendo palavras que começam com uma determinada letra do alfabeto. Exemplo: lista de todas as palavras do texto que começam com a letra A. Palavras não devem aparecer repetidas nas listas. Além disso, para cada palavra deverá ser criada uma lista informado a(s) linha(s) do texto na(s) qual(is) ela aparece. Um programa de testes deverá ser implementado, de maneira a receber um texto qualquer, criar o TAD Dicionário para o mesmo e, em seguida, permitir que determinadas operações desse TAD possam ser executadas por um usuário.

A estrutura de dados a ser utilizada para a implementação das listas lineares será OBRIGATORIAMENTE LISTA ENCADEADA (em todos os lugares em que uma lista for implementada).

Abaixo estão descritos, em detalhes, quais TADs deverão ser implementados neste trabalho, com os dados que devem estruturar e suas operações. É esperado que os alunos façam implementações que sigam completa e corretamente estas especificações.

1) TIPO ABSTRATO DE DADOS PALAVRA

Será necessário que você implemente o TAD Palavra. Cada elemento do tipo Palavra deverá ser representado por uma cadeia de caracteres (a palavra efetivamente) e uma lista linear que informa as linhas em que a palavra apareceu (Obs.: esta será apenas uma lista de números inteiros. Um TAD para esse tipo de lista pode ser implementado para deixar o código ainda mais organizado). As operações deste TAD devem ser, pelo menos:

- Cria Palavra vazia
- Preenche cadeia de caracteres
- Retorna cadeia de caracteres
- Imprime cadeia de caracteres
- Imprime Palavra (cadeia de caracteres + TAD Lista de Números de linha)

2) TIPO ABSTRATO DE DADOS LISTA DE PALAVRAS

Este TAD representa uma lista linear de elementos do tipo Palavra. Cada item da lista deve ser do tipo Palavra. Além da lista encadeada com itens do tipo Palavra, esse TAD deve conter um campo que contabiliza o número de palavras da lista. As operações disponíveis devem ser:

- Cria nova Lista de Palavras
- Insere nova Palavra (no final)
- Remove Palavra (dada a palavra)
- Remove Palavra (do final)
- Verifica Palavra (dada a palavra, verifica se já existe na lista)
- Retorna número de palavras
- Imprime

3) TIPO ABSTRATO DE DADOS DICIONÁRIO

O TAD dicionário deverá conter todas as palavras obtidas a partir de um texto de entrada. Para tanto, você deverá criar uma lista de palavras para cada letra do alfabeto, sendo uma corresponde à letra A, outra à letra B e assim sucessivamente (Obs: crie apenas as listas necessárias, isto é para letras que efetivamente aparecem iniciando palavras no texto). Por exemplo, se a palavra “amizade” apareceu no texto, ela deve ser inserida em uma lista de palavras do texto que começam com a letra 'a' (não é necessário diferenciar maiúsculo e minúsculo). Para cada palavra dessa lista, teremos uma lista das linhas em que ela ocorreu no texto, como, por exemplo, 1, 10, 25.

As operações que esse TAD deve disponibilizar são:

- a) Inicializa Dicionário – essa operação apenas cria um dicionário vazio
- b) Constrói Dicionário – recebendo um texto como parâmetro, cria o dicionário para as suas palavras
- c) Exibe lista de palavras do texto que começam com determinada letra do alfabeto. Essa operação deverá exibir todas as palavras registradas no TAD Dicionário que começam com a referida letra.
- d) Mostra todas as palavras do TAD Dicionário, na sequência alfabética.

SISTEMA DE TESTE

Para testar a implementação dos seus TADs, você deverá implementar um programa principal que utiliza o TAD Dicionário. Este programa deverá receber um texto como um arquivo de entrada, ler todas as palavras deste texto e inseri-las nas listas do Dicionário. Considere que o texto não possuirá pontuação nem linhas em branco. Além disso, todas as letras serão usadas na forma minúscula. Todas as palavras estarão separadas umas das outras por um espaço em branco.

TESTES

Um arquivo texto será disponibilizado para os alunos para realização de testes durante o desenvolvimento. Além desse, um outro arquivo será utilizado no dia das entrevistas para avaliação dos trabalhos.

Em particular, atente para:

- ☐ O programa deve ser organizado em módulos, conforme estudado em sala de aula. O módulo do programa principal deve estar separado dos módulos que compõem os TADs.
- ☐ O programa deve estar bem indentado e comentado
- ☐ Caso apareçam números fixos no código, estes devem ser definidos como constantes
- ☐ **Trabalhos copiados serão penalizados.**

ATENÇÃO: Soluções que não correspondam à implementação de Tipos Abstratos de Dados serão duramente penalizadas por não atenderem à especificação.

O que deve ser entregue:

- Todo código fonte produzido, incluindo os arquivos de cabeçalho
- Uma pequena porém completa documentação, descrevendo o objetivo do trabalho, o projeto do sistema implementado, as principais decisões de projeto, os módulos desenvolvidos, os métodos implementados e a conclusão. O formato para a entrega da documentação deve ser pdf
- Fazer um zip ou tar.gz de todos os arquivos, nomeá-lo com o nome e número de matrícula de todos, bem como número do TP, e submetê-lo apenas uma vez pelo PVANet Moodle (NÃO UTILIZAR COMPACTAÇÃO RAR!).

Como será a avaliação:

- Entrevista com os monitores
- Avaliação do código + documentação (monitores e professora)