

Exercícios

Listas lineares

Pilhas

1. Considere uma pilha em alocação sequencial. Escreva as funções para empilhar um elemento na pilha e desempilhar um elemento da pilha.
2. Considere uma pilha em alocação encadeada sem cabeça. Escreva as funções para empilhar um elemento na pilha e desempilhar um elemento da pilha.
3. Considere uma pilha em alocação encadeada com cabeça. Escreva as funções para empilhar um elemento na pilha e desempilhar um elemento da pilha.
4. Considere o problema de decidir se uma dada sequência de parênteses e chaves é bem-formada. Por exemplo, a sequência abaixo:

(() { () })

é bem-formada, enquanto que a sequência:

({) }

é malformada.

Suponha que a sequência de parênteses e chaves está armazenada em uma cadeia de caracteres s . Escreva uma função *bem_formada* que receba a cadeia de caracteres s e devolva 1 se s contém uma sequência bem-formada de parênteses e chaves e devolva 0 se a sequência está malformada.

5. As expressões aritméticas podem ser representadas usando notações diferentes. Costumeiramente, trabalhamos com expressões aritméticas em notação *infixa*, isto é, na notação em que os operadores binários aparecem entre os operandos. Outra notação frequentemente usada em compiladores é a notação *posfixa*, aquela em que os operadores aparecem depois dos operandos. Essa notação é mais econômica por dispensar o uso de parênteses para alteração da prioridade das operações. Exemplos de expressões nas duas notações são apresentados abaixo.

notação infix	notação posfixa
$(A + B * C)$	$A B C * +$
$(A * (B + C) / D + E)$	$A B C + * D / E +$
$(A + B * C / D * E - F)$	$A B C * D / E * + F -$

Escreva uma função que receba uma cadeia de caracteres contendo uma expressão aritmética em notação *infixa* e devolva uma cadeia de caracteres contendo a mesma expressão aritmética em notação *posfixa*. Considere que a cadeia de caracteres da expressão *infixa* contém apenas letras, parênteses e os símbolos $+$, $-$, $*$ e $/$. Considere também que cada variável tem apenas uma letra e que a cadeia de caracteres *infixa* sempre está envolvida por um par de parênteses.



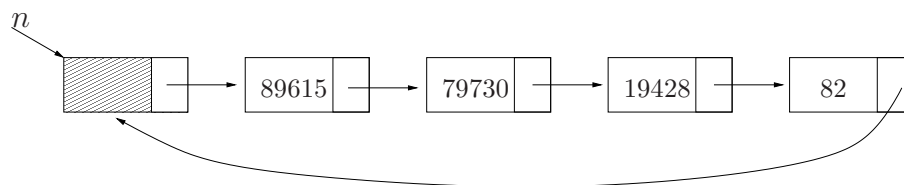
Filas

6. Considere uma fila em alocação sequencial. Escreva as funções para enfileirar e desenfileirar um elemento da fila.
7. Escreva as funções de enfileiramento e desenfileiramento em uma fila em alocação encadeada sem cabeça.
8. Escreva as funções de enfileiramento e desenfileiramento em uma fila em alocação encadeada com cabeça.
9. Considere uma pilha P vazia e uma fila F não vazia. Utilizando apenas os testes de fila e pilha vazias, as operações **Enfileira**, **Desenfileira**, **Empilha**, **Desempilha**, e uma variável **aux** do tipo **celula**, escreva uma função que inverta a ordem dos elementos da fila.
10. Implemente uma fila usando duas pilhas.
11. Implemente uma pilha usando duas filas.

Listas circulares

12. Escreva funções que implementem as operações básicas de busca, inserção e remoção sobre uma lista linear circular em alocação sequencial. Escreva duas funções para a operação de inserção: uma que insere um elemento sempre no final da lista e outra que insere um elemento em uma lista ordenada.
13. Escreva funções que implementem as operações básicas de busca, inserção e remoção sobre uma lista linear circular sem cabeça em alocação encadeada. Escreva três funções para a operação de inserção: uma que insere um elemento no final da lista; uma que insere um elemento no início da lista; e outra que insere um elemento em uma lista ordenada.
14. Escreva funções que implementem as operações básicas de busca, inserção e remoção sobre uma lista linear circular com cabeça em alocação encadeada. Escreva três funções para a operação de inserção: uma que insere um elemento no final da lista; uma que insere um elemento no início da lista; e outra que insere um elemento em uma lista ordenada.
15. Escreva uma função para liberar todos os nós de uma lista linear circular em alocação encadeada para a lista de espaços disponíveis da memória. Escreva duas versões dessa função, uma para uma lista linear circular com cabeça e outra para uma lista linear circular sem cabeça.

16. O **problema de Josephus** foi assim descrito através do relato de Flavius Josephus, um historiador judeu que viveu no primeiro século, sobre o cerco de Yodfat. Ele e mais 40 soldados aliados estavam encurralados em uma caverna rodeada por soldados romanos e, não havendo saída, optaram então por suas próprias mortes antes da captura. Decidiram que formariam um círculo e, a cada contagem de 3, um soldado seria morto pelo grupo. Josephus foi o soldado restante e decidiu então não se suicidar, deixando essa história como legado. Podemos descrever um problema mais geral como segue. Imagine n pessoas dispostas em círculo. Suponha que as pessoas são numeradas de 1 a n no sentido horário e que um número inteiro m , com $1 \leq m < n$, seja fornecido. Começando com a pessoa de número 1, percorra o círculo no sentido horário e elimine cada m -ésima pessoa enquanto o círculo tiver duas ou mais pessoas. Escreva e teste uma função que resolva o problema, imprimindo na saída o número do sobrevivente.
17. Um computador permite representar números inteiros até um valor máximo, que depende da sua organização e arquitetura interna. Suponha que seja necessário, em uma dada aplicação, o uso de números inteiros com precisão muito grande, digamos ilimitada. Uma forma de representar números inteiros com precisão ilimitada é mostrada na figura abaixo.



A lista linear circular com cabeça em alocação encadeada acima, armazena o número inteiro 82194287973089615. O tipo célula desta lista linear circular, não sofre alterações na sua definição, sendo o mesmo o tipo nó previamente definido nas aulas. Observe apenas que uma chave contém um número inteiro de até 5 algarismos.

Escreva uma função que receba duas listas lineares circulares contendo dois números inteiros de precisão ilimitada e devolva uma lista linear circular contendo o resultado da soma desses dois números.

Listas duplamente encadeadas

18. Escreva funções para implementar as operações básicas de busca, inserção e remoção em uma lista linear duplamente encadeada sem cabeça. Escreva três funções para a operação de inserção: uma que insere um elemento no final da lista; uma que insere um elemento no início da lista; e outra que insere um elemento em uma lista ordenada.



19. Escreva funções para implementar as operações básicas de busca, inserção e remoção em uma lista linear duplamente encadeada com cabeça. Escreva três funções para a operação de inserção: uma que insere um elemento no final da lista; uma que insere um elemento no início da lista; e outra que insere um elemento em uma lista ordenada.
20. Listas lineares duplamente encadeadas também podem ser listas circulares. Basta olhar para uma lista linear duplamente encadeada e fazer o ponteiro para o nó anterior do nó cabeça apontar para o último nó e o ponteiro para o próximo nó do último nó apontar para cabeça. Escreva funções para implementar as operações básicas de busca, inserção e remoção em uma lista linear duplamente encadeada circular com cabeça.
21. Escreva um procedimento que recebe uma lista linear duplamente encadeada e troca o primeiro elemento com o último, o segundo com o penúltimo e assim sucessivamente.