

**ATIVIDADE 3 - ESOFT - ESTRUTURA DE DADOS I - 52/2023****Período:**12/06/2023 08:00 a 30/06/2023 23:59 (Horário de Brasília)**Status:**ABERTO**Nota máxima:**0,50**Gabarito:**Gabarito não está liberado!**Nota obtida:****1ª QUESTÃO**

A estrutura "lista", apesar de ser parecida com as estruturas de pilha e fila, não inclui regras de entrada e saída, como as regras do tipo FIFO e LIFO. Todavia, as listas possuem características próprias que lhe dão grande versatilidade. É possível, também, criar pilha e fila dinâmicas. Para isso, você deve criar uma lista dinâmica e adicionar as regras LIFO e FIFO.

Fonte: OLIVEIRA, P. M.; PEREIRA, R. L. **Estrutura de Dados I**. Maringá: UniCesumar, 2019.

Observe a estrutura a seguir, que representa os nós de uma lista:

```
typedef struct no{
    int valor;
    struct no *proximo;
}No;
```

Considere o procedimento a seguir:

```
void procedimento(No **lista, int num){
    No *aux, *no= NULL;
    if(*lista != NULL){
        if((*lista)->valor == num){
            no = *lista;
            *lista = no->proximo;
        }else{
            aux = *lista;
            while(aux->proximo != NULL && aux->proximo->valor != num)
                aux = aux->proximo;
            if(aux->proximo != NULL){
                no= aux->proximo;
                aux->proximo = no->proximo;
            }
        }
    }
    if(no != NULL){
        free(no);
    }
}
```

Explique o procedimento anterior e informe para que serve em uma lista.

ALTERNATIVAS

- Primeiramente é definida a estrutura de Nós da lista a ser percorrida. (Contém o valor a ser salvo e o ponteiro *proximo que aponta para o próximo endereço do Nó a ser criado);
- a função procedimento recebe dois argumentos como parâmetros - um ponteiro *lista do tipo No que aponta para o endereço de início da lista e outra variável num do tipo inteiro a ser buscada.
- if(*lista != NULL) - serve para testar que o ponteiro *lista é diferente de NULL, que não é vazio ou que possui a