

## Exercícios - Lista Simplesmente Encadeada

1. Crie um menu para executar as operações sobre uma lista.
2. Crie as seguintes funções:

```
//verifica se duas listas são iguais
```

```
bool igual(No **lista1, No **lista2)
```

```
//Insere um valor na lista de forma ordenada
```

```
void insereOrdenado(No **lista, int valor)
```

```
//Insere um valor no final da lista
```

```
void insereFinal(No **lista, int valor)
```

```
//retorna o total de elementos da lista
```

```
int total(No **lista)
```

```
//retorna o valor do dado armazenado em um nó da lista, com base na posição passada por parâmetro
```

```
//o valor deve ser retornado por referência
```

```
//A função deve retornar true caso a leitura foi realizada com sucesso e false caso contrário
```

```
bool lePosicao(No **lista, int posição, int *valor)
```

```
//Remove um elemento da lista com base na posição do elemento passada por parâmetro
```

```
bool removePosicao(No **lista, int posicao)
```

```
//União de duas listas
```

```
void uniao(No **lista1, No **lista2, No **listaSaida)
```

3. Faça uma função que verifica se uma Lista L1 está ordenada (em ordem crescente ou decrescente).
4. Considere uma lista encadeada L1 representando uma sequência de caracteres. Construa uma função para imprimir a sequência de caracteres da lista L1 na ordem inversa.

**Exemplo:** Para a lista L1={A,E,I,O,U}, a função deve imprimir “UOIEA”.

5. Construa uma função que recebe como parâmetros uma Lista L1 de valores inteiros e um valor **X**. A função deve retirar os primeiros **X** valores da lista L1, inserindo-os no fim de L1. Use as funções de inserção e remoção separadas.

**Exemplo:**

L1: [3,5,8,9,12,11,7,10]

x: 4

L1 após a função: [12,11,7,10,3,5,8,9]

6. Um grupo de pesquisa em segurança da informação está estudando a frequência de uso de senhas semelhantes pelos usuários de um determinado sistema. Assim, foi criada uma lista contendo as senhas dos usuários. Uma das análises a ser realizada é a verificação de senhas não seguras. Uma senha pode ser considerada “não segura” se ela possuir uma quantidade muito pequena de caracteres ou for utilizada por uma quantidade muito grande de usuários. Para classificar as senhas não seguras, você deve criar uma lista simplesmente encadeada contendo todas as senhas com menos de 4 caracteres ou cuja frequência de utilização é maior que 5 (ou seja, senhas que aparecem mais que cinco vezes na lista). Crie a função `naoSeguras` que recebe uma lista de senhas e retorna outra lista contendo todas as senhas não seguras existentes.