

**LISTA DE EXERCÍCIOS PARA ESTUDO****Assunto: Listas Ligadas**

Para testar as funções abaixo, usem as funções dadas em aula (Material de Aula – Parte 2): para a inserção de elementos na lista e para imprimir os elementos da lista. Façam as adaptações para o tipo do elemento, quando for o caso.

**Níveis de Dificuldade – ND:** considerando o conhecimento exclusivo da disciplina  
1: fácil, 2: médio, 3: médio+, 4: difícil, 5: difícil e trabalhoso

**I - Sobre análise dos elementos da lista ligada**

1. Dada uma lista ligada. Construir uma função que conta o número de nós dessa lista. O número de nós encontrado deverá retornar pelo comando return. (ND 1)
2. Dada uma lista ligada contendo números inteiros **ordenados**. Construir uma função que verifica se há algum número repetido na lista ou não. Caso exista alguma repetição retornar 1 senão retornar 0. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 1)
3. Dada uma lista ligada contendo números inteiros **desordenados**. Construir uma função que verifica se há algum número repetido na lista ou não. Caso exista alguma repetição retornar 1 senão retornar 0. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 2)
4. Dada uma lista ligada contendo números inteiros. Construir uma função que verifica se a lista está ordenada ou não. Se ordenada retornar 1 senão retornar 0. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 1)
5. Dada uma lista ligada contendo números inteiros quaisquer. Construir uma função que verifica qual o número que se repete mais vezes nessa lista. Retornar a porcentagem de repetições desse número em relação à lista toda. Não há restrição quanto ao uso de outras estruturas, apenas não "estrague" a lista dada. (ND 3)

**II - Sobre a Criação de lista ligada:**

1. Construir uma função que cria uma lista ligada para conter em cada nó um número inteiro com a seguinte regra: o 1º elemento deverá ser igual a 1, o 2º é igual ao primeiro\*2, o 3º igual ao segundo \*2, e assim por diante. Parar quando o cálculo do elemento for maior do que 1000. Não inserir o resultado maior do que 1000. (ND 3)
2. Construir uma função que cria uma lista ligada para conter em cada nó um termo da seguinte soma: (ND 3)

$$\frac{1}{1} + \frac{3}{2} + \frac{5}{3} + \frac{7}{4} + \dots + \frac{99}{50}$$

3. Construir uma função que cria uma lista com 50 elementos, onde cada elemento da lista é um termo da seguinte série: (ND 3)

$$\frac{1!}{1} - \frac{2!}{3} + \frac{3!}{5} - \frac{4!}{7} + \frac{5!}{9} - \frac{6!}{11} + \frac{7!}{13} - \frac{8!}{15} + \dots$$

Isto é, o termo:  $\frac{1!}{1}$  é o primeiro elemento da lista;  $-\frac{2!}{3}$  é o segundo,  
e assim por diante.

4. Dada uma lista ligada contendo números inteiros (entre 1 e 10), não ordenados, e que há várias repetições de cada um deles. Construir uma função que cria uma segunda lista ligada para conter: no 1º nó a quantidade de repetições do número 1, no 2º nó as repetições do número 2, e assim por diante, até o 10º nó. (ND 3)

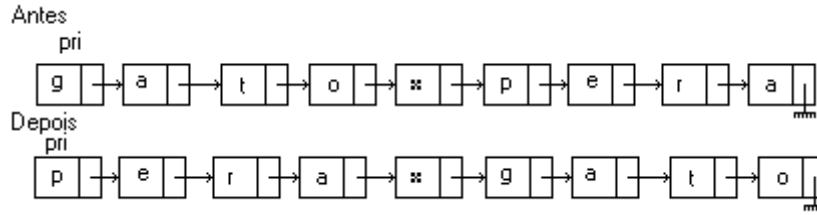
### III - Sobre Inserções na lista:

1. Dada uma lista ligada contendo números inteiros. Construir uma função que insere uma nova informação V após o nó que possui a informação armazenada em X. Supor X e V valores lidos na *main* e passados na lista de parâmetros. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 2)
2. Dada uma lista ligada contendo números inteiros. Construir uma função que insere após cada nó, um novo nó, com a mesma informação dele. No final a lista deverá estar com todos os valores duplicados. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 3)
3. Dada uma lista ligada contendo em cada nó um caractere referente à uma palavra. Construir uma função que insere entre cada caractere da palavra, um nó com o caractere '\*'. Não usar estruturas auxiliares para a solução, tais como, vetor ou outra lista ligada. (ND 3)

### IV - Sobre Rearranjo das ligações numa lista ligada :

1. Dada uma lista ligada contendo números inteiros. Construir uma função que troca de lugar o nó armazenado no endereço P com o nó armazenado no endereço S. Sabendo-se que o nó do endereço P precede o nó de endereço S, na lista ligada. (Não significa que o endereço P é menor do que S). Prever todas as possibilidades, por exemplo: os nós estarem nas pontas, os nós estarem ligados, existirem nós entre eles, existirem nós antes de P e/ou após S. A troca deverá ser feita, obrigatoriamente, fazendo-se o rearranjo das ligações, não criar novos nós ou mexer com o campo de informação. (ND4)
2. Dada uma lista ligada contendo números inteiros quaisquer. Construir uma função que ordena os elementos da lista ligada, apenas fazendo o rearranjo das ligações, sem mexer com o campo de informação ou criando novos nós. Sugestão: chame a função criada no exercício anterior quando precisar trocar de lugar dois nós. (ND4)
3. Dada uma lista ligada contendo, em cada nó um caractere, referente aos caracteres de duas palavras e, um nó com um '\*' separando as duas palavras. Construir uma função que troca de lugar os nós referentes à primeira palavra com os da segunda. Veja um exemplo a seguir. O algoritmo deve ser genérico para quaisquer duas palavras e deve ser feito apenas através do rearranjo das ligações, sem mover as informações e sem criar novos nós. (ND4)

Exemplo:



5. Dada uma lista ligada contendo números inteiros quaisquer. Construir uma função que reorganiza a lista colocando os nós com números pares no início da lista e os ímpares no final. O algoritmo deve ser feito apenas através do rearranjo das ligações, sem mover as informações e sem criar novos nós. (ND4)
6. Dada uma lista ligada contendo números inteiros ordenados. Construir uma função que separa os elementos da lista ligada ordenada em duas. A original fica com os pares e a nova com os ímpares. (ND3)
7. Dadas duas listas ligadas contendo números inteiros ordenados, uma apontada por **Pria** e a outra por **Prib**. Construir uma operação que une essas duas em uma também ordenada. Essa operação não deve fixar o elemento de uma delas e compará-lo com todos da outra. Levar em consideração que elas estão ordenadas, e à medida que os elementos vão sendo comparados, decisões devem ser tomadas. (ND4)
8. Dada uma lista ligada. Construir uma função que inverte a ordem das ligações dessa lista. Isto é, o último será o primeiro, o penúltimo o segundo e assim por diante. O algoritmo deverá ser feito apenas fazendo o rearranjo das ligações, sem mexer com as informações ou criando novos nós. (ND5)

#### V - Sobre Eliminação de nó da lista ligada:

1. Dada uma lista ligada contendo números inteiros. Construir uma função que elimina apenas o último nó. Verificar se a lista não está vazia. (ND2)
2. Dada uma lista ligada contendo números inteiros. Construir uma função que elimina o primeiro nó da lista e coloca no seu lugar o último nó da lista. O último nó deverá ser "desligado" do final e "ligado" no início. Não gere novo nó e nem troque as informações entre os nós. **Somente faça o rearranjo dos link's**. (ND3)
3. Dada uma lista ligada contendo números inteiros. Construir uma função que elimina todos os nós com conteúdo par. (ND3)
4. Dada uma lista ligada contendo números inteiros. Construir uma função que elimina o nó seguinte a um nó, contendo a informação armazenada num certo X, lido. Se X não existir na lista ou não existir o nó seguinte ao de X, retornar zero pelo *return*. Caso exista X e o seguinte, eliminar o seguinte e retornar 1 pelo *return*. (ND4)
5. Dada uma lista ligada contendo, **em cada nó, um caractere** referente à uma frase. Construir uma função que elimina todos os nós que possuem a vogal 'a'. (ND4)