**TRABALHO DE ESTRUTURAS DE DADOS**

**“Resta Um”**

**Usar: Listas Duplamente Encadeadas**

Em uma lista duplamente encadeada, remover nós da posição N até que a lista tenha apenas 1 nó. Para exemplificar, considere a lista encadeada abaixo:

NULL<->9<->8<->7<->6<->5<->4<->3<->2<->1<->NULL e

N == 2 (dado de entrada)

*Na 1ª. Passagem, o nó a ser removido será o 2 da lista, ou seja, o nó que contém o valor 8. A lista ficará:*

NULL<->9<->7<->6<->5<->4<->3<->2<->1<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 7. Com isso, o próximo nó a ser removido será o nó com 6, veja:*

NULL<->9<->7<->5<->4<->3<->2<->1<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 5. Com isso, o próximo nó a ser removido será o nó com 4, veja:*

NULL<->9<->7<->5<->3<->2<->1<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 3. Com isso, o próximo nó a ser removido será o nó com 2, veja:*

NULL<->9<->7<->5<->3<->1<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 1, porém a lista terminará, então deve-se retornar ao início dela... Com isso, o próximo nó a ser removido será o nó com 9, veja:*

NULL<->7<->5<->3<->1<->NULL e

*..e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 7. Com isso, o próximo nó a ser removido será o nó com 5, veja:*

NULL<->7<->3<->1<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 3. Com isso, o próximo nó a ser removido será o nó com 1, veja:*

NULL<->7<->3<->NULL e

*...e a contagem recomeça a partir do próximo ao removido, neste caso o nó com valor 7. Como a lista chegou ao fim, volta-se ao início dela e, com isso, o próximo nó a ser removido será o nó com 3, veja:*

NULL<->7<->NULL

Assim, o algoritmo termina e deve ser mostrado que o nó que restou é o nó que contém o valor 7.

OBS. A cada remoção, imprimir a lista para poder acompanhar o comportamento do algoritmo.

Bom trabalho.

Prof. Mauricio Duarte