

Lista 11 - Listas Encadeadas Parte 2

1. Crie a função `remove_tudo`, responsável por remover todas as células da lista encadeada. Lembre-se que, com exceção da célula-cabeça, todas as células foram criadas fazendo uso de alocação dinâmica, e portante requerem o uso da função `free`. Teste o seu código.
2. Provavelmente, a sua implementação de `remove_tudo` é iterativa. Crie uma nova função `remove_tudoR` com uma implementação recursiva. Teste o seu código.
3. Escreva em C/C++ uma função denominada `conte_celulas`, que recebe como parâmetro uma lista encadeada e retorna a quantidade de células que a lista possui. Teste o seu código.
4. Implemente uma função que receba duas listas encadeadas como parâmetro e retorne uma nova lista encadeada que consiste na fusão das duas listas (não é necessário realizar cópia das listas passadas como parâmetro). Teste o seu código.
5. Escreva uma função que encontre uma célula de conteúdo mínimo. Faça duas versões: uma iterativa e uma recursiva.
6. Uma lista é *crescente* se o conteúdo de cada célula não é maior que o conteúdo da célula seguinte. Escreva uma função que faça uma busca em uma lista crescente. Faça versões para listas com e sem cabeça. Faça uma versão recursiva e outra iterativa.