



Questão 01 - Conforme visto na aula anterior, implemente uma lista sequencial em C ou C++ para armazenar valores inteiros e que tenha os seguintes métodos:

- `push_front (int valor)` - Insere um valor na frente da lista e retorna 1 para sucesso ou -1 para falha;
- `push_back(int valor)` - Insere um valor no fim da lista e retorna 1 para sucesso ou -1 para falha;
- `insert(int index, int valor)` - Insere um valor em uma determinada posição da lista (index) e retorna 1 para sucesso ou -1 para falha;
- `pop_front()` - Exclui um valor da frente da lista retorna 1 para sucesso ou -1 para falha;
- `pop_back()` - Exclui o último elemento da lista retorna 1 para sucesso ou -1 para falha;
- `removeAt(int index)` - Remove o elemento localizado na posição passada com parametro e retorna 1 para sucesso ou -1 para falha;
- `sizeOf()` - retorna o tamanho atual da lista;
- `find(int valor)` - Busca um elemento na lista e retorna o index do elemento encontrado ou -1 para falha;
- `delete()` - Remove todos os elementos da lista retorna 0 caso a lista esteja vazia ou -1 para falha;
- `sort()` - Ordena de forma crescente a lista (use um dos algoritmos de ordenação vistos na unidade anterior);
- `reverse()` - Ordena de forma decrescente a lista (use um dos algoritmos de ordenação vistos na unidade anterior);

A lista deve ser implementada a partir de um ponteiro alocado dinamicamente e o tamanho da lista deve ser passada no início do programa. Não utilize soluções prontas da linguagem de programação. Você pode utilizar registros (structs) ou classes para implementar sua lista. Na função `main()` crie uma lista e teste todos os métodos desenvolvidos. Utilize os nomes dos métodos conforme a lista acima.