

1. Desenvolva um sistema de gerenciamento de estoque para uma loja de conveniência. O sistema deve permitir que o usuário adicione novos produtos ao estoque e remova-os quando necessário. Além disso, o sistema deve ser capaz de recuperar informações sobre um produto específico rapidamente.

Implemente uma **pilha** para rastrear as adições e remoções mais recentes de produtos, e uma **tabela hash** para armazenar informações detalhadas sobre cada produto, como **nome**, **preço** e **quantidade em estoque**. Como parte da solução, utilize o método de multiplicação para implementar a tabela hash e resolva colisões com **listas encadeadas** em um **vetor**.

2. Considere que você precisa armazenar os contatos de uma lista telefônica, onde cada contato é composto por um **nome** (*string*) e um **número de telefone** (*string*). O programa deve ser capaz de adicionar novos contatos, remover contatos existentes e buscar por um contato específico.

Para resolver esse problema, utilize **três estruturas de dados** diferentes:

- a. **Vetor**: armazena os contatos em uma lista não ordenada.
 - b. **Lista encadeada**: armazena os contatos em uma lista ordenada por nome.
 - c. **Árvore binária de busca**: armazena os contatos em uma estrutura de dados mais eficiente para busca.
3. Crie uma solução para um problema de análise sintática utilizado em compiladores para a formação de um dicionário de palavras-chave. O objetivo é ler uma lista de palavras-chave de um **arquivo de entrada** e armazená-las em um **dicionário** implementado como uma **árvore binária de busca**. Em seguida, o programa deve ser capaz de verificar se uma palavra está presente no dicionário ou não.

Considere que cada palavra-chave pode conter apenas letras minúsculas e que o arquivo de entrada contém uma palavra-chave por linha. As palavras-chave são lidas de um arquivo de entrada chamado "**palavras_chave.txt**" e armazenadas em uma árvore binária de busca. O programa permite ao usuário buscar uma palavra específica no dicionário. A comparação de palavras é feita de forma *case-insensitive*, ou seja, não faz diferença entre maiúsculas e minúsculas.