

Sets

Coleção não-ordenada de elementos **distintos**. Essa coleção pode estar vazia. Basicamente, são como conjuntos matemáticos.

Como representar: escrevendo os elementos ou uma propriedade comum

Operações: interseção, checar pertencimento, união.

Formas de Implementar: vetor de bits para representar um subconjunto do conjunto universal ou usando uma lista.

Dictionaries

Permitem operações comuns em coleções dinâmicas de dados, tais como busca, adição e exclusão de itens.

Como representar: uso simples de arrays (ordenados ou não) até técnicas mais avançadas, como hashing e árvores de busca balanceadas.

Operações: busca por um item, a adição de um novo item e a exclusão de um item da coleção.

Formas de Implementar: Pode-se usar desde estruturas mais simples, como arrays, até técnicas mais avançadas, como hashing. A eficiência da implementação muitas vezes depende do equilíbrio entre a eficiência da busca e as eficiências das operações de adição e exclusão.

M7

SETS, DICTIONARIES AND HASHING.

Hashing

Hashing é uma maneira eficiente de implementar dicionários em ciência da computação. Dicionários são estruturas de dados abstratas que representam conjuntos com operações de busca (lookup), inserção e exclusão definidas em seus elementos. Em prática, dicionários frequentemente lidam com registros que contêm campos diversos, como registros de estudantes em uma escola, registros cidadãos em um órgão governamental ou registros de livros em uma biblioteca.

Como representar: A implementação de hashing envolve a distribuição de chaves entre um array unidimensional chamado de tabela de hash. Essa distribuição é realizada por meio de uma função chamada função de hash, que atribui a cada chave um endereço na tabela.

Operações: As operações principais em hashing incluem a busca por uma chave, a inserção de uma nova chave e a exclusão de uma chave da coleção. A eficiência dessas operações depende da escolha adequada da função de hash e do tamanho da tabela.

Formas de Implementar: Existem duas abordagens principais para resolver colisões em hashing: Open Hashing (também conhecido como Separate Chaining) e Closed Hashing (também conhecido como Open Addressing). Na Open Hashing, as chaves são armazenadas em listas encadeadas ligadas às células da tabela de hash. Na Closed Hashing, todas as chaves são armazenadas diretamente na tabela, e diferentes estratégias, como sondagem linear e double hashing, podem ser usadas para resolver colisões. A escolha entre essas abordagens depende das características específicas do problema e dos requisitos de eficiência