

Trabalho de AEDS: Vector

Por: David Francisco Cordeiro Neto, João Pedro de Almeida Dupim e Pedro Henrique Santos

Sumário

01

Introdução

02

**Pontos
Principais**

03

Como Utilizar

04

**Exemplo
Prático**

01

Introdução ao Vector

A estrutura

Vector faz parte do conjunto de estruturas da Standard Template Library e representa um vetor dinâmico, ou seja, um vetor que não precisa ter um tamanho fixo pré-definido.

Vector é uma estrutura que armazena elementos de maneira sequencial, permitindo fácil acesso aos elementos por meio de índices. Ele faz parte da biblioteca padrão do C++ e fornece uma alternativa mais flexível e conveniente aos arrays tradicionais em C.



02

Pontos Principais

A principal característica do Vector é que ele pode redimensionar automaticamente para acomodar um número variável de elementos. Isso significa que você não precisa especificar o tamanho do vetor antecipadamente, ao contrário de um array clássico.

Os vetores oferecem acesso rápido aos elementos por meio de índices. Isso permite um acesso direto aos elementos por meio de operações de índice, garantindo um desempenho eficiente. São compatíveis com uma ampla gama de algoritmos disponíveis na biblioteca padrão do C++. , permitindo a aplicação direta desses algoritmos aos elementos armazenados no vetor.

Principais Operações oferecidos pela biblioteca

push_back

Adiciona um elemento no final do vetor.

pop_back

Remove o último elemento do vetor.

insert

Insere elementos em uma posição específica no vetor.

front

Acesso ao primeiro elemento do vetor.

back

Acesso ao último elemento do vetor.

swap

Troca o conteúdo de dois vetores.

03

Como Utilizar

Para utilizar a biblioteca vector em C++, primeiro, é necessário incluir o cabeçalho `<vector>` no código. Depois disso, é possível criar instâncias de vetores para armazenar e manipular elementos dinâmicos.

```
#include <iostream>
#include <vector>
```

A biblioteca oferece uma ampla gama de funcionalidades para manipular e gerenciar coleções de elementos dinâmicos de maneira eficiente. Ela proporciona uma flexibilidade valiosa, permitindo a adição, remoção, acesso e modificação de elementos de forma dinâmica

04

Exemplo Prático