

Resumo-LP1

Caio Vinicius

April 2023

Contents

1	Pilha	3
2	Fila	4
3	Lista	5
4	Vector	6
5	Deque	7
6	Set	8
7	Map	9

1 Pilha

O primeiro dos containers existentes em c++ que serão citados aqui. A Pilha tem como principal característica que todos os seus elementos que são colocados nela são os primeiros a serem removidos. Dessa forma, é como se cada termo da Pilha fosse empilhado e apenas pudesse mexer com o que está no "topo", para acessar os outros teria que remover o último inserido.

Um exemplo de utilização da Pilha em sistemas é em funções recursivas. Cada vez que ocorre o chamado de uma função recursiva dentro dela mesma essa nova chamada é colocada no topo de uma Pilha e apenas ela será rodada naquele momento, as outras ficam em espera até que a do "topo" termine de rodar e seja eliminada.

2 Fila

Agora irei falar sobre a Fila. A Fila é, de certa forma, oposta a Pilha, isso porque agora o mecanismo de remoção muda. Na Fila o primeiro elemento a ser retirado dela será o que entrou primeiro, mas a sua inserção não muda, os elementos continuam sendo inseridos no final dela.

Dentre as suas aplicações podemos citar o controle da comunicação para impressão. Por exemplo, se for preciso em uma empresa imprimir vários documentos sendo que cada documento vem de máquinas diferentes, será preciso algo que indique a ordem de impressão, assim, a fila se torna útil, pois se todos tiverem a mesma prioridade, será impresso o documento que foi enviado primeiro para a impressora.

3 Lista

Falarei agora sobre a Lista. Em c++, a Lista é uma estrutura de dados que cria uma lista duplamente encadeada, isso é, tanto a sua inserção, quanto a sua eliminação de elementos pode ser feita tanto no início quanto no final dela. Além disso, todos os seus elementos são acessáveis facilmente, não importando eles serem os primeiros ou os últimos na estrutura.

Além de poderem ser aplicadas da mesma forma que os containers anteriores, a Lista também pode ser aplicada no uso de gerenciamento de memória, indicando os espaços para alocar memória e os que devem ser liberados.

4 Vector

Talvez o mais conhecido de todos os containers vistos, o vector é criado já sendo um array dinâmico, ou seja, ele pode mudar de tamanho conforme a necessidade. Ele é capaz de ser usado como todas as outras estruturas de dados já vistas até agora, porém ele é mais fácil de ser manipulado, aceitando operações de vetores comuns.

Conforme dito sobre sua capacidade de adaptação de tamanho, o vector é muito usado em aplicações que buscam esse aspecto, dentre elas, pode ser citado a programação voltada a objetos.

5 Deque

Parecido com a Lista, o Deque tem a capacidade de adicionar e eliminar os elementos tanto do seu início quanto no seu final. Além disso, como o Vector, é possível acessar seus membros facilmente usando [] ou iteradores.

Ele consegue ser aplicado em praticamente qualquer área da programação, já que, possui características de todos as outras estruturas de dados citadas anteriormente.

6 Set

Agora sobre o container Set. Bem parecido com um conjunto matemático, ao inserir elementos no Set eles são ordenados automaticamente (de forma crescente por padrão, mas há possibilidade de mudar para decrescente) e não aparecerá repetições. Cada elemento inserido ganha uma chave que é o seu próprio valor, assim, não é possível ter dois elementos com a mesma chave.

Dentre suas aplicações podemos citar a red-black-tree, uma árvore binária de busca balanceada. Ela é especificamente focada em busca binária.

7 Map

Por último, temos o container Map. Ele é parecido com o Set, mas eles possuem uma diferença crucial, que é a chave. Enquanto que no Set a chave é o próprio valor do que irá conter, no Map a chave pode ser qualquer uma, assim, o Map funciona por meio de pares, sendo um deles a chave especificada (que pode ou não ter o mesmo tipo do elemento) e o elemento que será guardado.

Dentre suas aplicações temos as mesmas do Set, além de operações como fazer CRUD em algum banco de dados, já que será necessário guardar informações específicas para cada pessoa.