

C++, STL e Complexidade

Lucas Santos Calumbi



Cronograma

- **C++**
 - **Estrutura e Sintaxe**
 - **Macros**
 - **Compilação e Execução**
- **STL**
 - **Vetores**
 - **Filas**
 - **Pilhas**
 - **Conjuntos**
 - **Dicionários**
- **Complexidade**
 - **Análise de Complexidade**
 - **Estimativa de Tempo**

C++

A linguagem de programação que domina o competitivo



Live Coding!!!

- Estrutura e Sintaxe
- Macros
- Compilação Execução

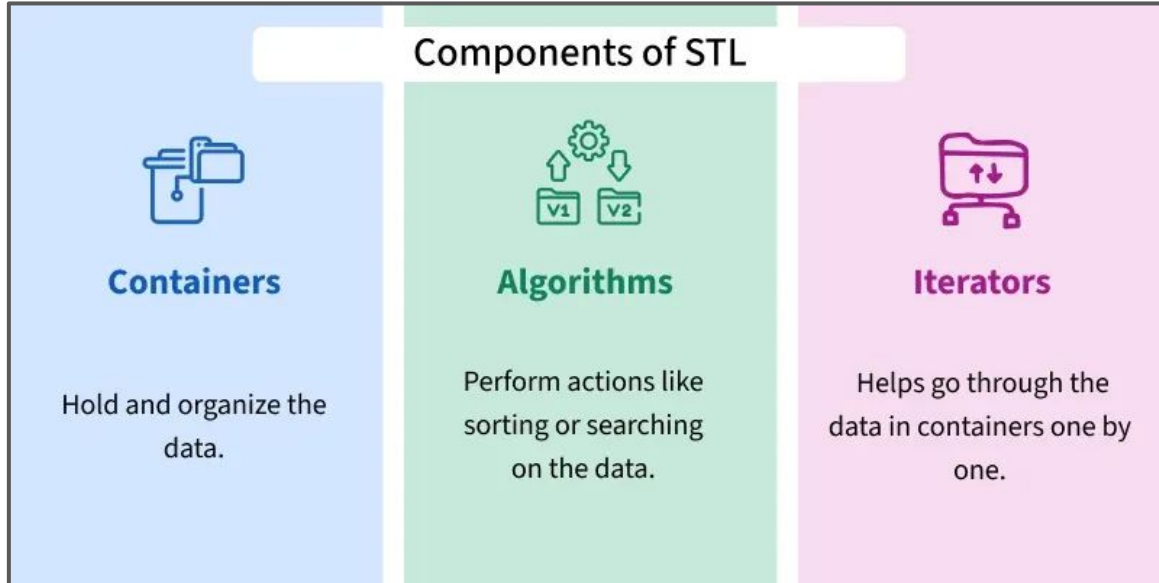
Questões no Beecrowd

- [Soma Simples - 1003](#)
- [Mês - 1052](#)
- [Tabuada - 1078](#)
- [Números Primos - 1165](#)

Standard Template Library (STL)

Standard Template Library (STL)

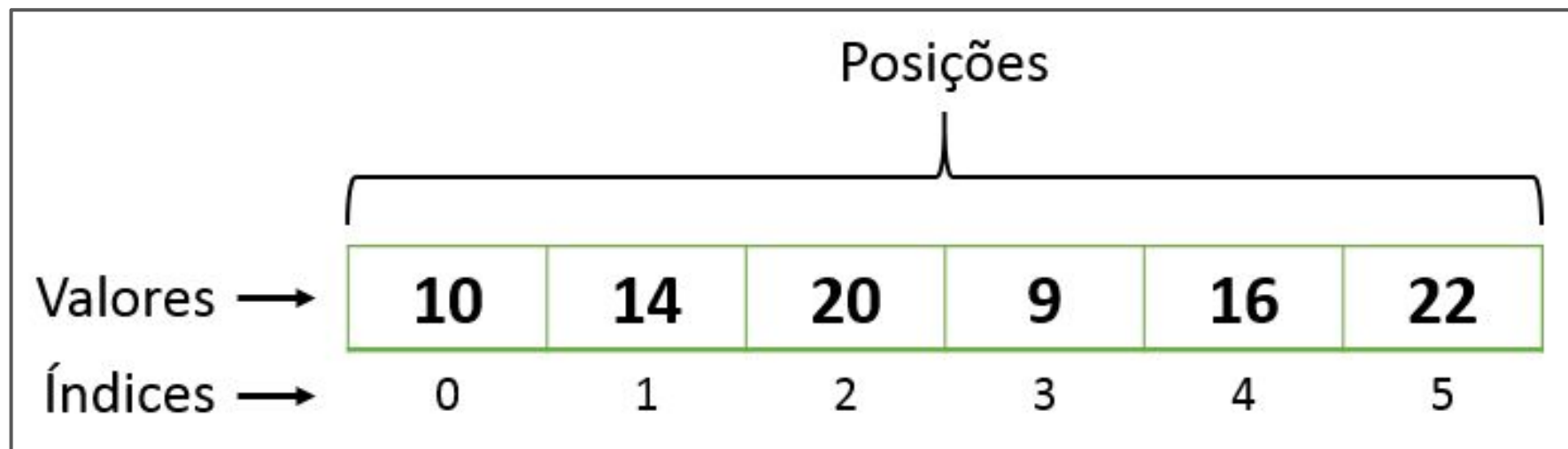
A STL é uma biblioteca padrão do C++ que contém, entre outras coisas, Estruturas de Dados e Algoritmos implementados e prontos para uso.



Hoje veremos:

- Vetores
- Filas
- Pilhas
- Conjuntos
- Dicionários

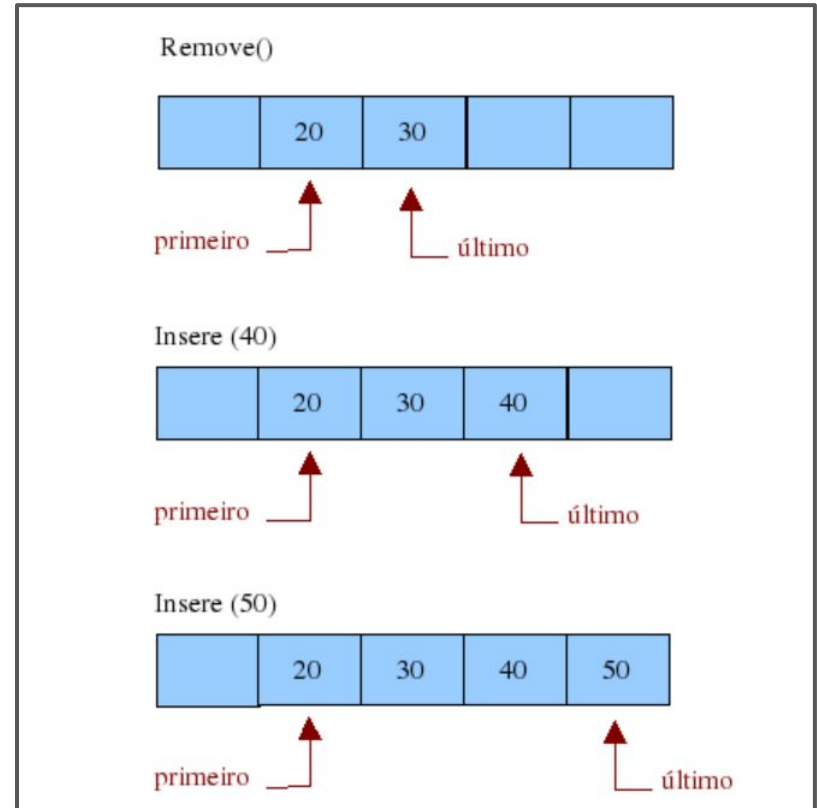
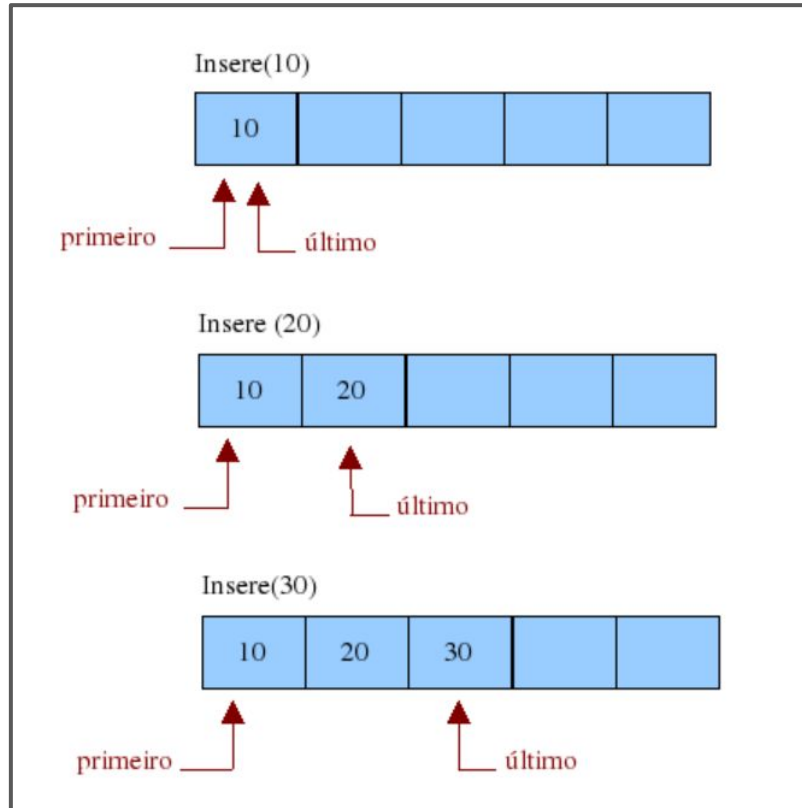
Vetor (vector)



Questão

- [Beecrowd: Menor e Posição - 1180](#)

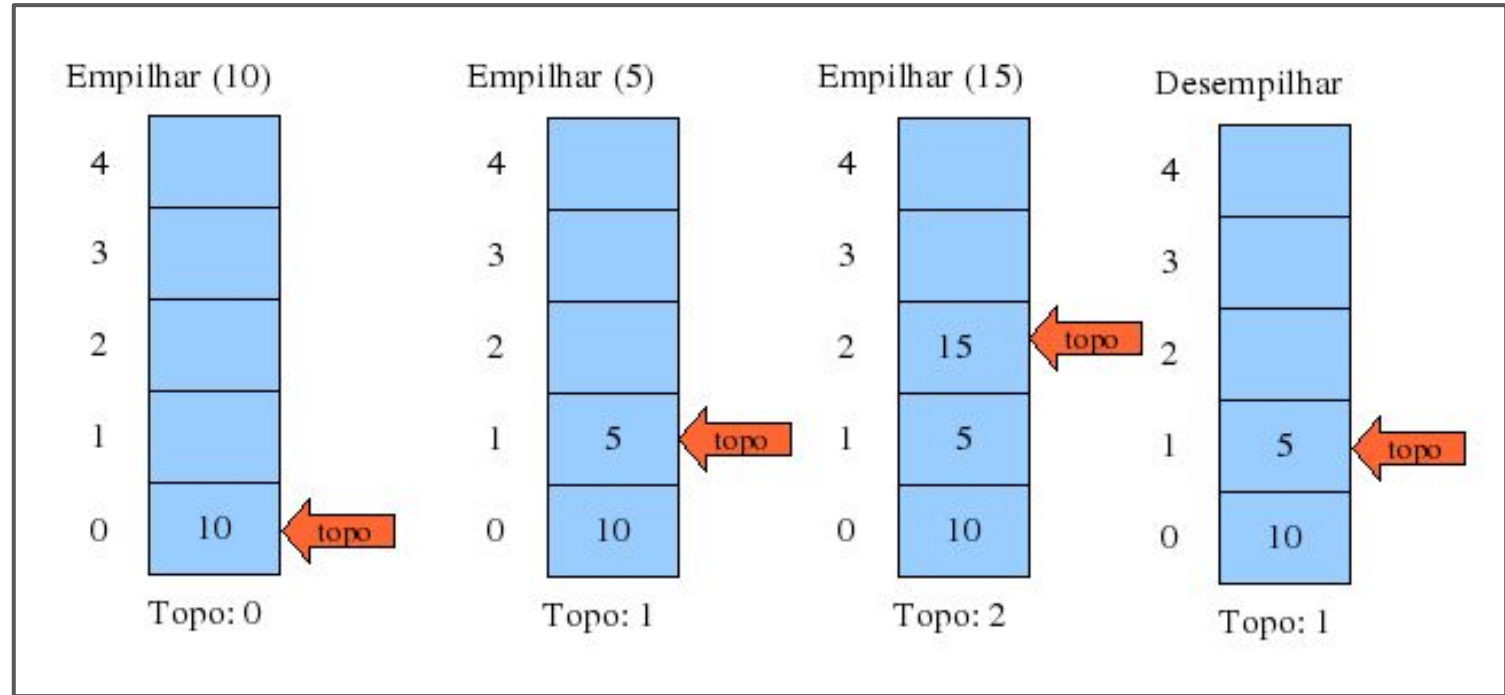
Fila (queue)



Questão

- [Beecrowd: Jogando Cartas - 1110](#)

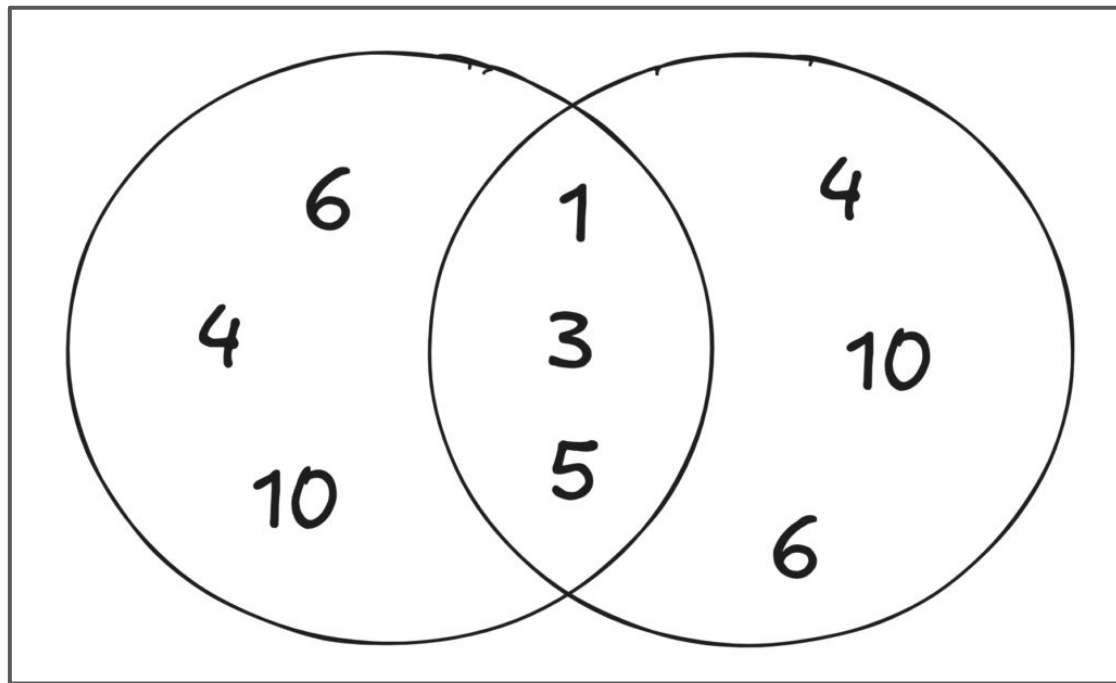
Pilha (stack)



Questão

- [Balanço de Parênteses I - 1068](#)

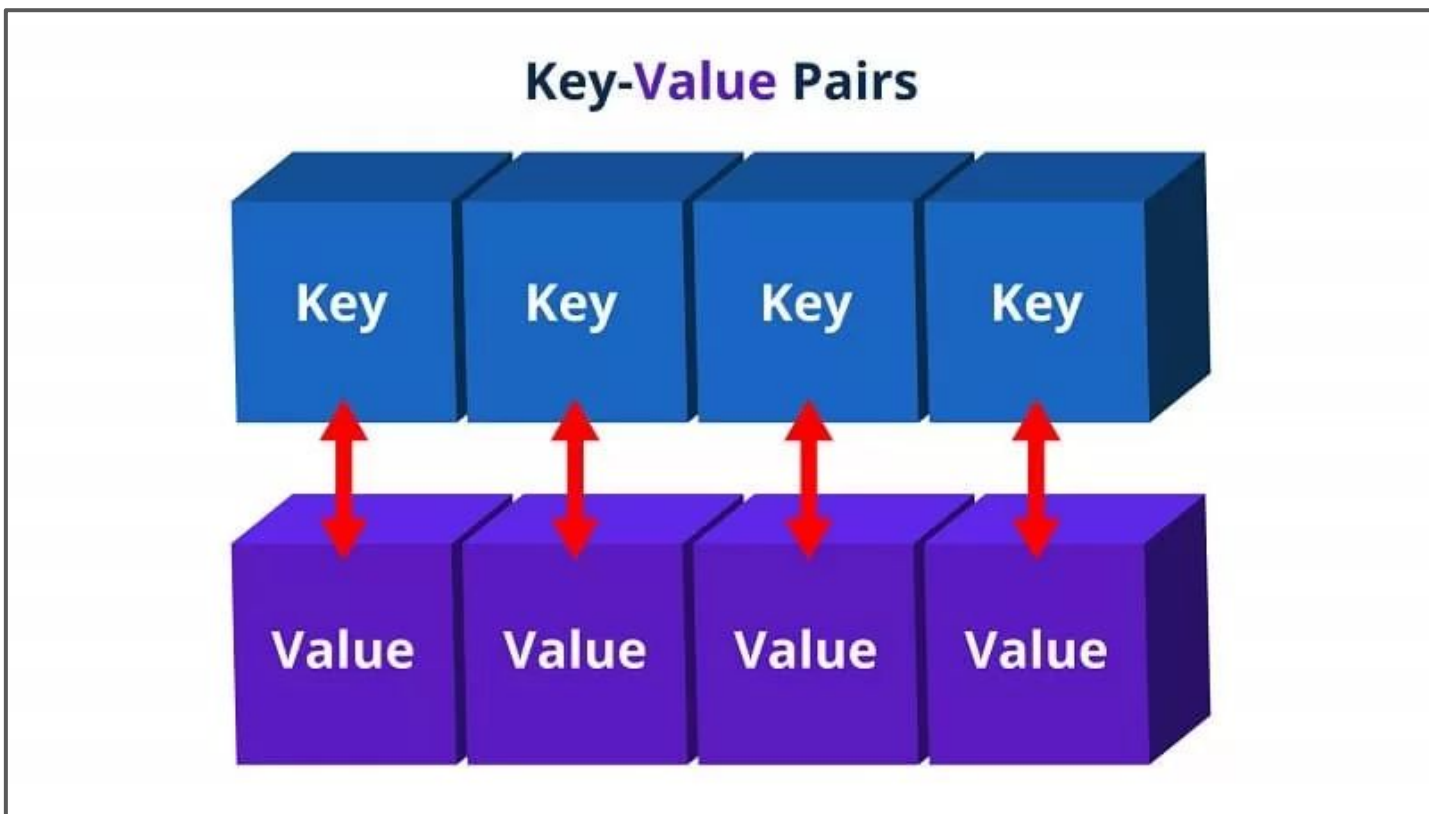
Conjunto (set)



Questão

- [CSES: Distinct Numbers - 1621](#)

Dicionário (map)



Problema

- Contagem de letras numa String

Dado uma string S, exiba em ordem alfabética a quantidade de vezes que cada letra aparece

OBS: se a quantidade for 0, não deve-se exibir a letra

Entrada	Saída
banana	a: 3 b: 1 n: 2

Análise de Complexidade e Notação Big O

Como não tomar TLE

O que é Complexidade?

Complexidade de algoritmos é o estudo de quanto um algoritmo “custa” para rodar, geralmente medido em:

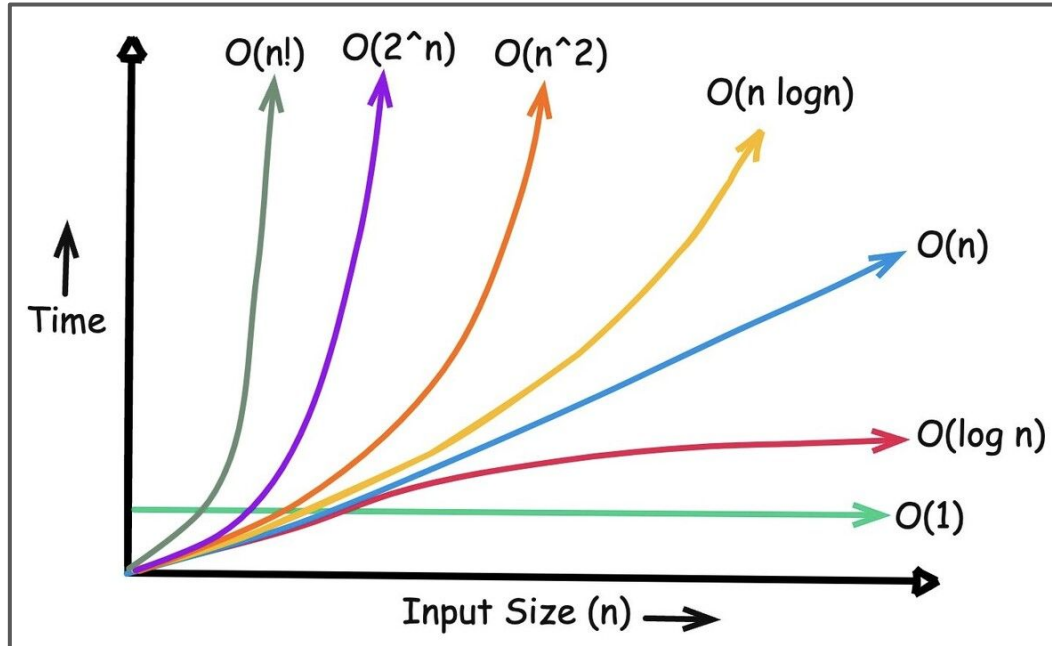
- Tempo: quanto o tempo de execução cresce conforme o tamanho da entrada aumenta.
- Espaço: quanta memória o algoritmo precisa.

Por que isso é importante?

Porque dois algoritmos que resolvem o mesmo problema podem ter desempenhos muito diferentes. **A complexidade permite prever se um algoritmo será rápido ou lento mesmo antes de implementá-lo.**

Notação Big O

A forma mais comum de expressar complexidade é a notação Big-O, que descreve o comportamento do algoritmo para entradas grandes:



Análise de Complexidade e Estimativa de Tempo

Problemas:

- Múltiplas consultas para verificar se um dado número está no vetor
- [CSES: Static Range Sum Queries - 1646](#)

Regra de ouro:

10^8 operações \approx 1 segundo

Questões:

- [Beecrowd: Fila do Recreio - 1548](#)
- [Beecrowd: Trilhos - 1062](#)
- [CSES: Maximum Subarray Sum - 1643](#)
- [CSES: Playlist - 1141](#)
- [CSES: Movie Festival - 1629](#)
- [CSES: Ferris Wheel - 1090](#)

Referências

- Steven Halim, Felix Halim, Suhendry Effendy – Competitive Programming 4 – Book 1.1 (2022)
- [GeeksforGeeks – Standard Template Library](#)
- [USP – BixeCamp 2025](#)
- [UFMG – Intro. à Programação Competitiva](#)