

Trabalho Prático I - Programação Orientada a Objetos



Alunos - Fabio Henrique Alves Fernandes | 19.1.4128
Igor Santiago Almeida Paixão | 19.1.4033



Professor - Guillermo Camara Chavez



Instituição - Universidade Federal de Ouro Preto | Departamento de Computação



Data De Entrega - 9 de novembro de 2021

Criação das Classes

Imóvel

Apartamento

Casa

Chácara

UML

Uso da STL no trabalho

Uso geral da STL

Acessar uma posição específica de um contêiner

Adicionar um elemento e manter somente elementos únicos no contêiner

Inserção ou remoção no final

Retornar um valor baseado em uma chave específica (não necessariamente inteiros)

Inserção e remoção no início

Busca por um elemento

Contêiner com o comportamento de primeiro a entrar e o último a sair

Contêiner com o comportamento de primeiro a entrar e o primeiro a sair

Criação das Classes

Para a implementação do nosso trabalho, criamos a classe principal Imóvel, e suas classes polimórficas Apartamento, Casa e Chácara

Imóvel

A classe imóvel tem como atributos:

- Proprietário;
- Valor;
- Rua;
- Bairro;
- Cidade;
- Número;
- Quantidade de banheiros;
- Quantidade de quartos.

A classe, além de seu construtor, de seu destrutor e de seus getters e setters, conta também com a sobrecarga virtual do método `out<<` para a saída de seus atributos. Também temos a sobrecarga dos operadores `<` e `>` para ajudar durante a ordenação.

Apartamento

A classe Apartamento tem como atributos, além de todos os pertencentes a Imóvel:

- Andar em que está localizado;
- Taxa de condomínio;
- A existência ou não de elevador;
- A existência ou não de sacada.

Além do construtor, do destrutor e dos getters e setters, temos a sobrecarga que completa a saída feita na classe base Imóvel.

Casa

A classe Casa tem como atributos, além de todos os pertencentes a Imóvel:

- Quantidade de andares;
- A existência ou não de sala de jantar.

Além do construtor, do destrutor e dos getters e setters, temos a sobrecarga que completa a saída feita na classe base Imóvel.

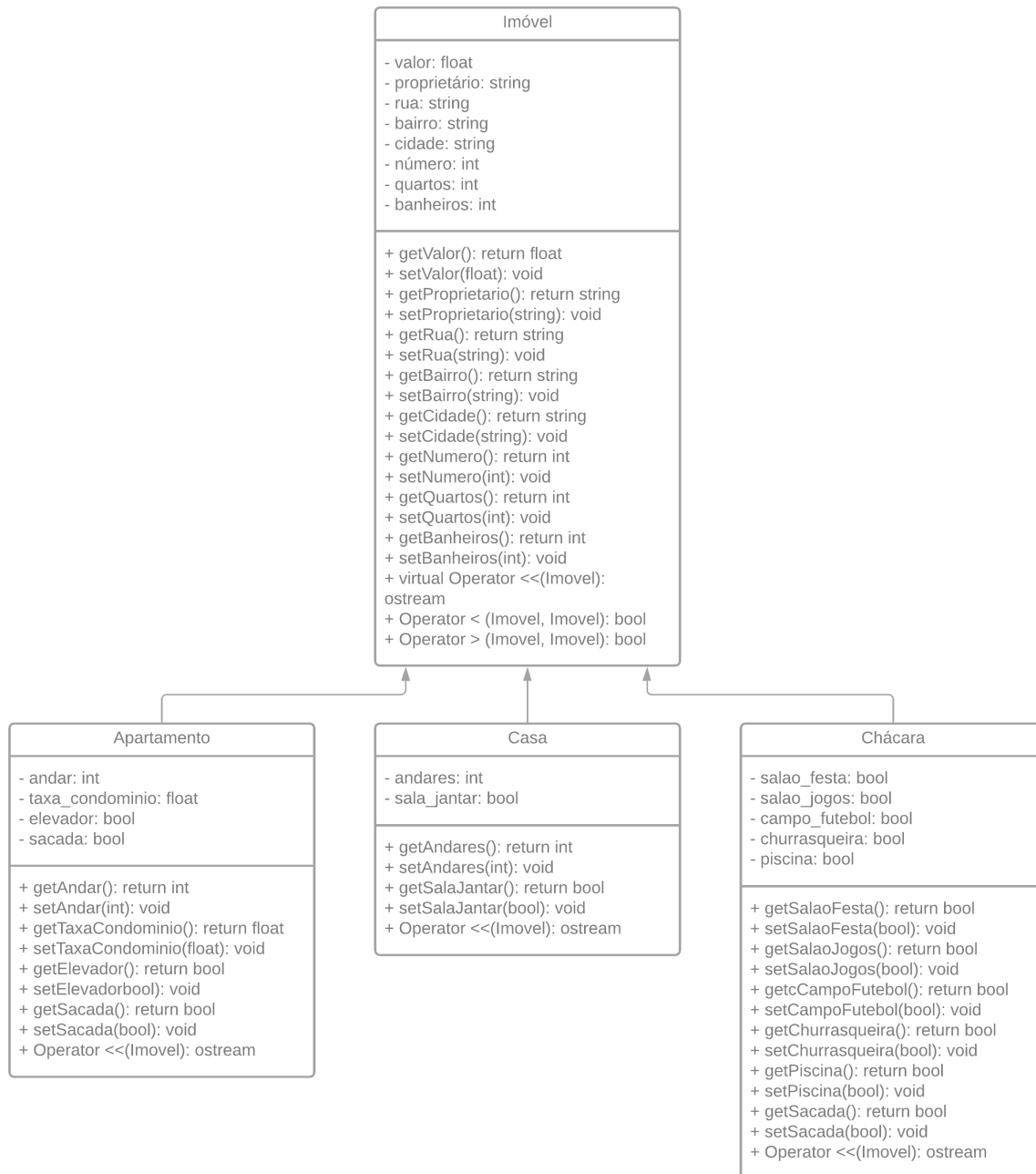
Chácara

A classe Chácara tem como atributos, além de todos os pertencentes a Imóvel:

- A existência ou não de salão de festas;
- A existência ou não de salão de jogos;
- A existência ou não de campo de futebol;
- A existência ou não de churrasqueira;
- A existência ou não de piscina.

Além do construtor, do destrutor e dos getters e setters, temos a sobrecarga que completa a saída feita na classe base Imóvel.

UML



Uso da STL no trabalho

Para o nosso sistema, usamos Vector para todas as questões para que não precisemos de conversões entre as STLs e pela facilidade em percorrer o Vector. Além de a função de ordenação funcionar bem.

Uso geral da STL

Acessar uma posição específica de um contêiner

Para o acesso a uma posição específica podemos usar o Vector ou um Array, já que temos a possibilidade de ter acesso usando o operador [] ou, no caso do Vector, usar o método `at()`, que retornam o elemento esperado.

Adicionar um elemento e manter somente elementos únicos no contêiner

Um dos contêineres que permitem a inserção somente de elementos únicos é o Set, que tem essa característica desde sua concepção.

Inserção ou remoção no final

Um dos contêineres que melhor faz a inserção ou a remoção no final é o Vector, já que o acesso a esse elemento é muito rápido ($O(1)$)

Retornar um valor baseado em uma chave específica (não necessariamente inteiros)

Nesse caso, Multiset é uma opção bem viável.

Inserção e remoção no início

O Deque é uma ótima opção, não só para a remoção/inserção no início, mas também para no final.

Busca por um elemento

Multimap é uma opção bem interessante quando queremos buscar um elemento.

Contêiner com o comportamento de primeiro a entrar e o último a sair

O contêiner com essa característica é basicamente a Stack (pilha), já que ela vai empilhando itens, de forma que o primeiro item que entrou ficou no final da pilha, já o último fica no topo.

Contêiner com o comportamento de primeiro a entrar e o primeiro a sair

O contêiner com essa característica é a Queue (fila), onde sempre que o primeiro elemento entrar, ele vai ser sempre o primeiro a sair.