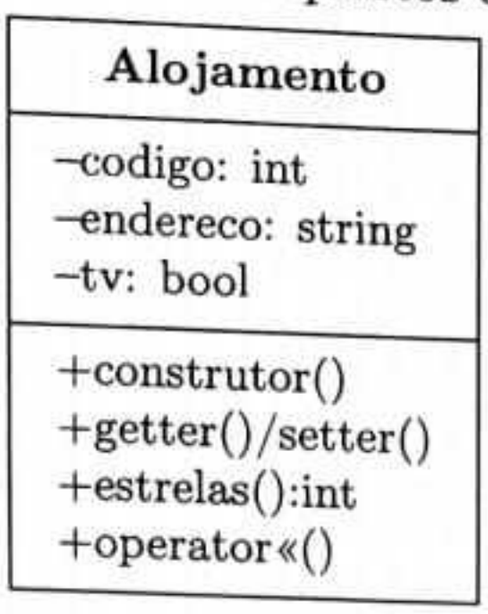
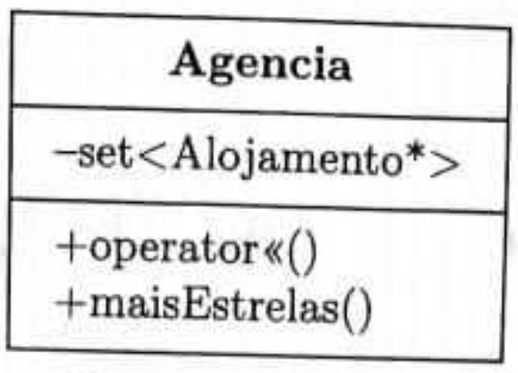
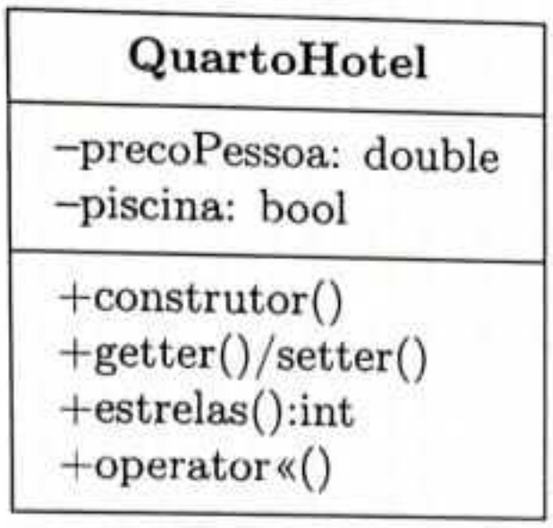
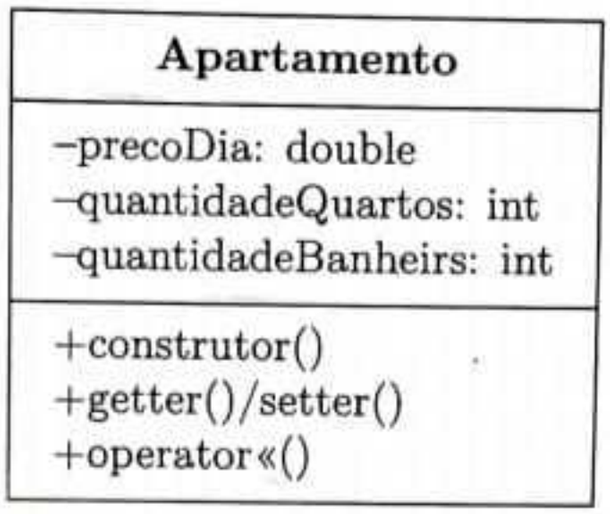
3. (8 pts) A continuação são mostradas algumas classe que fazem parte de um sistema de reserva de quartos de hotéis e apartamentos.









Realize as seguintes operações (**considere que os getter e setter de todas as classes encontram-se implementadas**):

(a) (2 pts) Implemente as classe **Alojamento**, e as classes concretas **QuartoHotel** e **Apartamento**. De cada classe, deve ser implementado:

i. A definição da classe;

ii. Construtor com todos os parâmetros; e

iii. Sobrecarga do operador «. As classes derivadas, além de mostrar os atributos próprios, devem mostrar os atributos da classe Base. Deve se evitar a redundância de código, não chamando os getters dos atributos da classe base na implementação da sobrecarga nas classes derivadas.

(b) (1.5 pts) Implementar o método **estrelas()**. Na classe Alojamento retorna 2 se o alojamento tem televisão, 1 em caso contrario. Na classe **QuartoHotel** retorna a quantidade de estrelas da classe **Alojamento** mais 1 se tiver piscina.

(c) (1.5 pts) Implementar a classe **Agencia**. Defina um *conteiner* do tipo *set*. O critério de ordenação dos dados é baseado no código do Alojamento. Na implementação também deve se incluir a codificação da função de comparação do set.

(d) (3 pts) Implemente o método **maisEstrelas()** da classe **Agencia** que retorna um *conteiner* de iteradores às Alojamentos com maior número de estrelas.

Universidade Federal de Ouro Preto - UFOP
Disciplina: BCC 221 - Programação Orientada a Objetos
Professor: Guillermo Cámara-Chávez

Aluno: _____ **No.** _____

A cola não será tolerada. Se alguém for pego colando, será reprovado com Zero. É considerado cola: olhar/copiar da prova de outro ou deixar outro aluno olhar sua prova. A interpretação faz parte da avaliação. Não são permitidas perguntas ou qualquer outro tipo de comentários durante a prova

1ra. Avaliação

1. (1 pt) Localize os erros em cada uma da seguintes sequências e explique como corrigi-lo(s)

   - Suponha que o seguinte protótipo é declarado na classe **Time**

     ```
     void ~Time();
     ```

   - Localize os erros na seguinte classe

     ```
     class Example{
         int data;
         static int count;
     public:
         Example(int y = 10):data(y){}
         int getIncrementedData() const{
             return data++;
         }
         static int getCount(){
             cout << "Data is " << data << endl;
             return count;
         }
     };
     ```

   - Suponha que o seguinte protótipo é declarado na classe **Employee**

     ```
     int Employee(const char*, const char*);
     ```

2. (1 pt) Determine se cada uma das instruções a seguir é verdadeira ou falsa. Se falsa, explique por quê

   (a) É perigoso referenciar um objeto de classe derivada com um *handle* de classe básica

   (b) Os parâmetros de *template* de uma definição de *template* de função são utilizados para especificar os tipos de argumentos para a função, especificar o tipo de retorno da função e declarar variáveis dentro da função

   (c) As palavras-chave *class* e *typename* tal como utilizadas como um parâmetro de tipo de *template* significam, especificamente, "qualquer tipo de classe definido pelo usuário".

   (d) A programação polimórfica pode eliminar a necessidade da lógica *switch*.