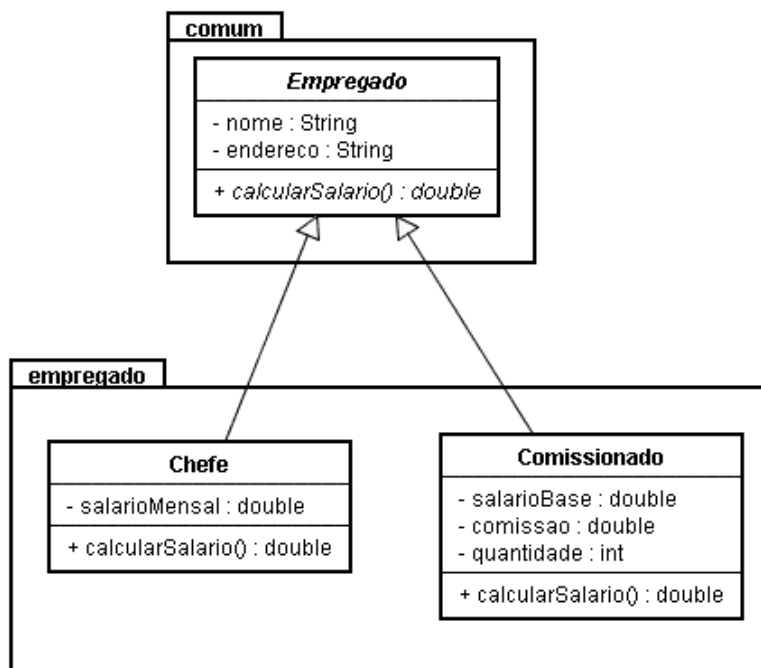


BACHARELADO EM TECNOLOGIA DA INFORMAÇÃO
DISCIPLINA: Linguagem de Programação I
PROFESSOR: PATRICK TERREMATTE

AVALIAÇÃO PRÁTICA – UNIDADE 2
Questão Prática
(6,0 pontos)

1. Considere o diagrama de classes em UML abaixo:



- [2,0 pontos] Implemente as classes acima em C++.
 - Note que a classe `Empregado` é uma classe abstrata, e que o método `calcularSalario()` da classe `Empregado` é um método abstrato.
 - Considere os modificadores de acesso dos atributos e métodos definidos no diagrama.
 - Considere que o método `calcularSalario()` da classe `Comissionado` retorna o valor da seguinte expressão: $(\text{salarioBase} + \text{comissao} * \text{quantidade})$.
 - Ajuste a sobrecarga dos métodos construtores, e informe qual o tipo de polimorfismo utilizado nos comentários.
- [1,0 ponto] Como preferir (usando `vector` ou `array`), implemente um vetor de chamado “departamento” que contenha pelo menos 2 chefes e 3 comissionados.
 - Dependendo da sua abordagem, informe nos comentários se o vetor está alocado no segmento de pilha (*stack*) ou na memória livre (*heap*).
- [1,0 ponto] Implemente uma forma de salvar o vetor de objetos `departamento` em arquivo binário usando `fstream`, em seguida recupere o vetor de objetos na variável de um novo vetor.

- [2,0 pontos] Altere a função abaixo para ordenar o vetor pelo salário calculado usando template.
 - Note que a função só ordena arranjos de elementos do tipo int. Adapte a função abaixo para funcionar como um *template* para qualquer tipo numérico, e informe qual o tipo de polimorfismo utilizado nos comentários do código.
 - Em seguida imprima os vetor de forma ordenada.

```
void ordenaPorSalario(int d[], int N){
    int i,j,aux,indMenor;
    for(i=0; i<N-1; i++){
        indMenor=i;
        for(j=i+1; j<N; j++){
            if(d[j]<d[indMenor]){
                indMenor=j;
            }
        }
        aux=d[i];
        d[i]=d[indMenor];
        d[indMenor]=aux;
    }
}
```