Exercicio3)

Construtores são rotinas em aplicações orientada a objetos para inicializar um objeto dentro da classe, podendo conter parâmetros ou não, possuem o mesmo nome do objeto da classe, não retornando valor.

Destrutores fazem o papel contrário aos construtores, fazendo a finalização do objeto, ou limpeza, para que não fique a aplicação aumentando o consumo de memória, são instanciados com um til (~) na frente do nome.

Exemplo:

Class Doenca{

Doenca();

~Doenca();

}

Doenca::Doenca()[

Doenca \*d = new Doenca();

}

DiagramaExercicio5)

Funcionario

cargo: string;

salario: double;

void setCargo(string);

void setSalario(double);

string getCargo();

double getSalario();

Pessoa

nome: string;

idade: int;

sexo: string;

void setNome(string);

void setIdade(int);

void setSexo(string);

string getNome();

int getIdade();

string getSexo();

Exercicio4)

Programação orientada a objetos (POO) é um **paradigma** que visa fornecer o **encapsulamento** para que classes e **membros** possam ser selecionados por meio da instanciação de classes quando o escopo for **public**.

Exercicio9)

STL é uma biblioteca que contém um conjunto de classes para C++, fornecendo estruturas de dados e funções de uso comum, como por exemplo listas, vetores, pilhas, etc.

Um exemplo de utilização é a classe vector, onde você declara uma estrutura desse e terá acesso a métodos próprios da classe para poder inserir, remover, buscar dados que foram armazenados nesta estrutura.

vector <string> nomeFuncionario;

nomeFuncionario.push\_back(“Leticia”);

nomeFuncionario.push\_back(“Bianca”);

nomeFuncionario.pop();

nomeFuncionario.size();

Exercicio10)

Sobre os principais conceitos do paradigma de Orientação a Objetos, pode se dar ênfase aos 4 pilares da POO, que são a Abstração, Polimorfismo, Herança e Encapsulamento.

Herança é quando duas ou mais classes “se comunicam” entre si, compartilhando uma relação entre classe mãe e classe filha(s), onde a classe mãe possui métodos e atributos que podem ser reaproveitados nas classe filhas. Para declarar uma herança:

Class Funcionario : Pessoa{}

Class mãe || class filha

Polimorfismo só existe quando há herança, onde acontece de uma classe filha ao herdar métodos e atributos da mãe, pode acontecer de que na filha seja declarado um método com mesmo nome, mesmo retorno mesma quantidade de parâmetros, igual como está na classe mãe, mas na classe filha a ação deste método é diferente da que é executado na classe mãe.

Encapsulamento é como se fosse uma “proteção” de atributos e métodos, mais utilizado para atributos, da classe para que não sejam acessados diretamente pelo usuário, por exemplo, um atributo saldo terá sua visibilidade como private para não poder ser acessado, então através de métodos, sacar e depositar, terá seu valor alterado pelos mesmos.

Abstração é a ideia de interpretar sobre um objeto de classe, ou seja, um levantamento de dados do que aquele objeto precisa ter para atender aos requisitos, ser considerado como tal, levantando suas características, que serão os seus atributos, e as suas ações, que será os métodos da classe.