- 2. Quais os três recursos característicos de LPs que suportam a Orientação a Objetos? A estruturação do código em classes e métodos permite implementar os principais conceitos da POO: O polimorfismo, a herança e o encapsulamento.
- 3. Qual a diferença entre classe e objeto? Classe é uma abstração que serve como "molde" para um objeto. Um objeto é uma instância de uma classe que possui os atributos e métodos definidos por ela.
- 4. Explique os diferentes níveis de proteção de atributos e métodos em Java e C++

Os diferentes níveis de proteção são :

- public : Atributos e métodos acessíveis a qualquer elemento do código
- protected : Atributos e métodos acessíveis a qualquer elemento do mesmo pacote
- private : Atributos e métodos acessíveis a qualquer elemento da mesma classe
- 5. Explique as diferentes formas de relacionamento entre classes

O relacionamento entre classes distintas podem ser de :

- Dependência: Quando uma classe requer outra classe para executar algum de seus métodos ou tem outra classe como algum de seus atributos, podemos dizer que ela depende da outra para existir.
- Generalização : Quando uma classe é uma expansão de outra classe, podemos dizer que a classe pai é uma forma generalizada de sua classe filho.

Em java o termo 'extends' é usado para denotar esses tipos de relação.

6. Defina os mecanismos de herança e herança múltipla.

A herança é um mecanismo que faz com que uma classe tenha acesso aos atributos e métodos de outra classe. A herança múltipla faz com que uma classe tenha acesso aos atributos e métodos de múltiplas outras classes.

7. Quais os problemas relacionados ao suporte a herança múltipla? Como Java e C++ resolvem estes problemas?

Os problemas da herança múltipla estão relacionados à ambiguidade no código, quando por exemplo duas classes pais possuem métodos com o mesmo nome, se uma classe derivada das duas tentasse chamar esse método, o compilador não saberá a qual dos métodos a classe filha está se referindo. Para lidar com isso :

- Java não implementa herança múltipla;
- C + + implementa herança múltipla, e resolve as ambiguidades através da clausula "using".
- 8. Explique as regras de compatibilidade entre classes.

Uma classe é compatível com todas as suas superclasses e com ela mesma, porém não é necessariamente compatível com suas subclasses.

9. Que mecanismo de suporte a LP deve ter para suportar esta regra de compatibilidade?

Para suportar essa regra, a linguagem de programação deve permitir que um objeto de uma classe 'X' seja tratado como um objeto de classe 'Y', uma vez que 'Y' seja derivado de 'X'.

10. O que é um método abstrato? O que é uma classe abstrata? Para que servem estes mecanismos?

Uma classe abstrata é uma classe que não pode ser instanciada e um método abstrato é um método declarado sem implementação Esses elementos servem para partilhar código entre classes muito relacionadas e são aconselhadas se as classes que são extensões das classes abstratas compartilhem muitos campos ou métodos, ou se elas tiverem modificadores de acesso não públicos.

## 11. Explique o conceito de interfaces de Java.

Uma interface é um recurso que faz com que um determinado número de classes tenham propriedades em comum. Utilizar interfaces garante que as classes contidas nela contenham as mesmas funções definidas pela interface embora elas possam ser implementadas de modo diferente em cada uma destas classes.