**Teste**

1. O que é programação procedural?

É um conjunto de procedimentos refinados que definem a estrutura global do programa. O programa termina quando acaba de chamar sua lista de procedimentos.

1. Qual vantagem a programação procedural tem em relação à programação não estruturada?

A programação procedural fornece a um programa uma estrutura global: dados e procedimentos. Os procedimentos também o ajudam a ver como deve programar uma tarefa. Em vez de escrever um único bloco de processamento grande, você divide os procedimentos em subprocedimentos. Os procedimentos fornecem um nível de reutilização. Você pode criar bibliotecas de procedimentos reutilizáveis.

1. O que é programação modular?

A programação modular, tenta melhorar algumas deficiências encontradas no programação procedural. A programação modular divide os programas em vários componentes ou módulos constituintes. Ao contrário da programação procedural, que separa dados e procedimentos, os módulos combinam os dois.

1. Quais vantagens a programação modular tem em relação à programação procedural?

A programação modular divide os programas em vários componentes ou módulos constituintes. Ao contrário da programação procedural, que separa dados e procedimentos, os módulos combinam os dois.

1. Liste uma deficiência da programação procedural e da programação modular.

Procedural: Como os dados e o procedimento são separados, não existe nenhum encapsulamento dos dados.

Modular: Os módulos não são extensíveis, ou seja, não permite alterações incrementais em um módulo sem abrir o código a “força” e fazer as alterações diretamente.

1. O que é programação orientada a objetos?

A POO dá o próximo passo lógico após a programação modular, adicionando herança e polimorfismo ao módulo. A POO estrutura um programa, dividindo-o em vários objetos de alto nível.

1. Quais são as seis vantagens e objetivos da programação orientada a objetos?
2. Natural
3. Confiável
4. Reutilizável
5. Manutenível
6. Extensível
7. **Oportuna**
8. Explique um dos objetivos da programação orientada a objetos.

O ciclo de vida do projeto de software moderno é frequentemente medido em semanas. A POO ajuda nesses rápidos ciclos de desenvolvimento. A POO diminui o tempo do ciclo de desenvolvimento, fornecendo software confiável, reutilizável e facilmente extensível.

O software natural simplifica o projeto de sistemas complexos. Embora você não possa ignorar o projeto cuidadoso, o software natural pode otimizar os ciclos de projeto, pois você pode se concentrar no problema que está tentando resolver.

**Quando você divide um programa em vários objetos, o desenvolvimento de cada parte pode ocorrer em paralelo. Vários desenvolvedores podem trabalhar nas classes independentemente. Tal desenvolvimento em paralelo leva a tempos de desenvolvimento menores.**

1. Defina os termos:
   1. Classe: Define os atributos e comportamentos comuns compartilhados por um tipo de objeto. Os objetos de certo tipo ou classificação compartilham os mesmos comportamentos e atributos. As classes atuam de forma parecida com um cortador de molde ou biscoito, no sentido de que você usa uma classe para cria ou instanciar objetos.
   2. Objeto: Uma construção de software que encapsula estado e comportamento. Os objetos permitem que você modele seu software em termos reais e abstrações. O objeto é uma instância de uma classe.
   3. Comportamento: É uma ação executada por um objeto quando passada uma mensagem ou em resposta a uma mudança de estado, ou seja, é algo que um objeto faz.
2. Como os objetos se comunicam entre si?
3. O que é um construtor?

Método usado para inicializar objetos durante sua instanciação. Você chama a criação de objetos de instanciação porque ela cria uma instância o objeto da classe.

1. O que é um acessor?

Dá acesso aos dados internos de um objeto. Entretanto os assessores ocultam o fato de os dados estarem em uma variável, em uma combinação de variáveis ou serem calculados. Os assessores permitem que você mude ou recupere o valor e têm ‘efeitos colaterais’ sobre o estado interno. (get)

1. O que é um mutante?
2. O que é this?

É uma referência que aponta para a instância do objeto. Cada objeto tem sua própria referência para si mesmo. A instância usa essa referência para acessar suas próprias variáveis e métodos.