

## **Definição de Construção de Software (CS)**

É a referência aos detalhes sobre a criação de um software funcional através da combinação de programação, testes unitários, testes de integração e debugging.

### **CS.1 Quais são os 5 Fundamentos de CS?**

- Minimizar Complexidade
- Antecipar Mudanças
- Construção para Verificação
- Reuso
- Padrões na Construção

### **CS.2 O que é refatoração?**

É o processo de modificar um sistema de software para melhorar a estrutura interna do código sem alterar seu comportamento externo.

### **CS.3 Considerações Práticas**

#### **1. O que é um Framework?**

Um projeto de software reutilizável que pode ser refinado e estendido para disponibilizar uma porção de uma funcionalidade, coleção de classes abstratas, concretas e interfaces entre elas

### **CS.4 Qual ou Quais dos 5 fundamentos de CS é ou são mais importantes? Desenvolva o porque.**

Os fundamentos mais importantes são Minimizar complexidade, Reuso e Padrões na Construção, pois minimizando a complexidade evita de dar erros inesperados nos testes, o reuso diminui o tempo programando algo que já está pronto e padrões na construção evita que cada um construa do seu jeito, tudo fica padronizado e quem vier de fora consegue entender mais facilmente.

### **CS.5 Como você diferenciaria complicado de complexo? De um exemplo.**

Complicado é uma coisa difícil de entender e complexo é uma coisa que você tem como prever o que vai acontecer.

Exemplo: Montar um quebra cabeça de mil peças é complicado, mas se estiver faltando uma peça se torna complexo pois não tem como prever que não poderá ser montado e nem onde está a peça faltante.

## **CS.6 Testes Automatizados auxiliam em quais fundamentos de CS?**

[X] Minimizar Complexidade [X] Antecipar Mudanças [X] Construção para Verificação [X] Reuso [ ] Padrões na Construção

## **CS.7 O que é reuso? Como podemos alcançar reuso na construção de software?**

Refere-se a utilizar um recurso existente para solucionar um problema diferente. Podemos alcançar na utilização de mesmas bibliotecas, módulos, códigos fontes, entre outros.

## **CS.8 O que é TDD? Você concorda ou discorda sobre TDD?**

Prática onde o desenvolvedor escreve testes antes da implementação. Concordo visando que cada parte do código será testado e desenvolvido, diferente de outras práticas que o teste é efetuado apenas no final.