**Acadêmico: Janderson Almeida dos Santos nº:2201324444**

**1. R-** <https://circuits.io/circuits/4025699-the-unnamed-circuit/edit>

**2. R-** <https://circuits.io/circuits/4043211-the-unnamed-circuit/edit>

**3. R-** <https://circuits.io/circuits/4044209-the-unnamed-circuit/edit>

**4. Defina o que é teste de software e descreva o microciclo do TDD.**

**R-** [Teste de software é o processo de execução de um produto para determinar se ele atingiu suas especificações e funcionou corretamente no ambiente para o qual foi projetado.](http://www.devmedia.com.br/introducao-a-testes-de-software-parte-1-definicoes-iniciais/16075) O seu objetivo é revelar falhas em um produto, para que as causas dessas falhas sejam identificadas e possam ser corrigidas pela equipe de desenvolvimento antes da entrega final.

**R-** TDD se escreve um pequeno teste, em seguida se escrever código de produção para que passe no teste, enquanto não quebra nenhum teste existente. Ele faz você decidir o que você deseja antes de construí-lo, fornecendo um feedback se tudo está funcionando com suas expectativas atuais.

As etapas do ciclo de TDD na lista a seguir são baseadas na descrição de Kent Beck, em seu livro Development Test-Driven:

1. Adicione um pequeno teste.

2. Execute todos os testes e veja o novo falhar, talvez nem mesmo compilar.

3. Faça as pequenas mudanças necessárias para o teste passar.

4. Execute todos os testes e ver o novo passar.

5. Refatore para remover duplicação e melhorar a sua expressividade.

**6. Faça uma pesquisa sobre três frameworks para teste de unidade e apresente as vantagens e desvantagens de cada um, bem como, uma tabela descrevendo os itens (assertivas, análise de memory leak e outros) suportados por cada um deles.**

**PHPunit**

* **Vantagens:** Permite a criação de códigos de testes com agilidade, Amplamente utilizado pela comunidade de código aberto.

**jUnit**

**Vantagens:**

* Permite a criação rápida de código de teste enquanto possibilita um aumento na qualidade do sistema sendo desenvolvido e testado;
* Não é necessário escrever o próprio framework;
* Amplamente utilizado pelos desenvolvedores da comunidade código-aberto, possuindo um grande número de exemplos;
* Uma vez escritos, os testes são executados rapidamente sem que, para isso, seja interrompido o processo de desenvolvimento;
* JUnit checa os resultados dos testes e fornece uma resposta imediata;
* Pode-se criar uma hierarquia de testes que permitirá testar apenas uma parte do sistema ou todo ele;
* Escrever testes com JUnit permite que o programador perca menos tempo depurando seu código;
* JUnit é LIVRE.

**Desvantagens:**

* O programador pode não adotar a idéia do uso da ferramenta para a melhora da qualidade do software, seja por causa da falta de costume ou da falta de interesse do mesmo;
* Se há pressão para a pronta entrega do sistema o programador ou testador podem “burlar” os Testes.

**xUnit**

**Vantagens:**

* É feito sobre um módulo (classe).