Testes servem para buscar erros em um software, fazendo-o falhar intencionalmente. Hoje, boas baterias de testes são vitais para garantir a boa qualidade do programa.

-> Testes também podem diminuir riscos.

-A metodologia de teste é complexa, devendo ser cuidadosamente planejada e monitorada.

* Testes dinâmicos: Com a aplicação pronta, em execução
* Testes estáticos: Análise do código-fonte em si, sem executá-lo. (Avaliado como um documento)

-> Testes devem ser feitos DURANTE TODO O CICLO DE VIDA DO SOFTWARE, MESMO APÓS SER COLOCADO EM PRODUÇÃO (manutenção do software)

-Testes podem, e devem, ser feitos em um ambiente similar ao de uso final. Testes servem para garantir que o software atende os requisitos do usuário e dos acionistas da companhia.

-O aumento da complexidade dos software trouxe a necessidade ainda maior de testes, para avaliar cenários inesperados; Por isso, é importante que **O TESTADOR SEJA INDEPENDENTE!**

--> Conceitos extras:

-Oráculo de testes: Diz qual o resultado esperado daquele teste

-Base de testes: Tudo que for utilizado para montar e rodar o teste

-Ciclos de teste: Processo de testes feito a cada *release* (entrega) do produto.

* Objetivos de testes:

-Avaliar produtos;

-Identificar e reportar defeitos

-Ter certeza que o programa atende os requisitos

-> Verificação: Verificar se as informações dos requisitos estão coesas;

-> Validação: Quando os requisitos estiverem fechados, verificar se o software os atende.

* Testes de componente: De pequenas partes do software
* Testes de aceite: Verificar se atende o que os stakeholders (partes interessadas) pediram;

-O teste descobre o defeito; A depuração vasculha o código para encontrá-lo e concertá-lo;

-> A depuração olha o código linha a linha para “entender” o defeito.

* Em teoria, testadores não fazem depuração, mas os testes e a confirmação do concerto.