

1.1.1 –

A- O que é :

Um compilador C.

Um compilador é um programa (ou um grupo de programas) que, a partir de um código escrito em uma linguagem, o código fonte, cria um programa semanticamente equivalente porém escrito em outra linguagem, código objeto.

Um compilador é um dos dois tipos mais gerais de tradutores, sendo que o segundo tipo que a ele deve ser comparado é um interpretador.

Normalmente, o código fonte é escrito em uma linguagem de programação de alto nível, com grande capacidade de abstração, e o código objeto é escrito em uma linguagem de baixo nível, como uma sequência de instruções a ser executada pelo processador.

O processo de compilação é composto de análise e síntese. A análise tem como objetivo entender o código fonte e representá-lo em uma estrutura intermediária. A síntese constrói o código objecto a partir desta representação intermediária.

A análise pode ser subdividida ainda em análise léxica, análise sintática e análise semântica. A síntese é mais variada, podendo ser composta pelas etapas de Geração de código intermediário, otimização de código e geração de código final (ou código de máquina), e somente esta última etapa é obrigatória.

Classicamente, um compilador traduz um programa de uma linguagem textual facilmente entendida por um ser humano para uma linguagem de máquina, específica para um processador e sistema operacional. Atualmente, porém, são comuns compiladores que geram código para uma máquina virtual que é, depois, interpretada por um interpretador.

Assembler

Montador. Este é o programa que transforma o código escrito na linguagem Assembly em linguagem de máquina, substituindo as instruções, variáveis pelos códigos binários e endereços de memória correspondentes. Os compiladores de várias linguagens de alto nível fazem a compilação dos programas em duas etapas, na primeira transformando o código fonte em código Assembly e em seguida gerando o binário com a ajuda de um Assembler.

Linker

Ligador (Vinculador no Brasil) ou Linker ou Linkeditor, em informática, é um programa que liga objectos gerados por um compilador, formando assim o ficheiro executável final. O trabalho de um Ligador é bastante simples: Estabelece a amarração ou vinculação entre referências mais abstratas para referências mais concretas. [1] Por exemplo, pega uma referência escrita por um programador como obtenhalmposto e estabelece a vinculação para a "localização de 726 bytes após o início do módulo executável no módulo iosys".

B- O que é um RTOS, descreva uma utilização.

Em geral, um sistema operacional (SO) é responsável por gerenciar os recursos de hardware e hospedar os aplicativos que são executados no computador. Um sistema operacional de

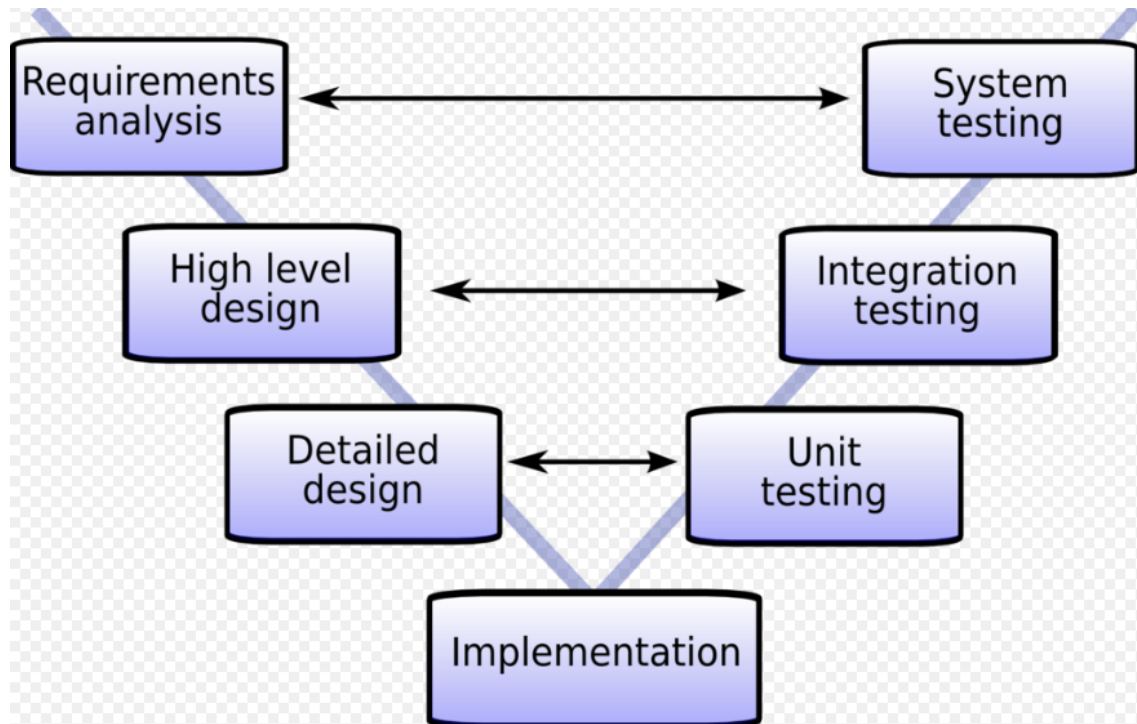
tempo real (RTOS) também executa essas tarefas, mas é projetado especialmente para rodar aplicações com extrema precisão e alto grau de confiabilidade. Isso pode ser especialmente importante para sistemas de medição e controle.

Os sistemas operacionais de tempo real foram projetados para trabalhar com resposta a eventos e com sistemas de controle em malha fechada. Um exemplo de aplicação com resposta a eventos, na qual é imprescindível a utilização de RTOS é um sistema de airbag automotivo, este sistema necessita de precisão e confiabilidade, pois a resposta a um estímulo deve ser feita em um determinado espaço de tempo. Já um exemplo de aplicação de controle em malha fechada em que o RTOS é necessário, é para o controle de velocidade em automóveis, pois também é necessário precisão e confiabilidade. Ambos os sistemas exigem a realização de uma operação dentro de um tempo determinado. Esse tipo de desempenho é chamado de determinístico.

C- O que é um desenvolvimento em V (modelo V)?

O Modelo V é um modelo conceitual de Engenharia de Sistemas/Desenvolvimento de Produto visto como melhoria ao problema de reatividade do modelo em cascata. Ele permite que, durante a integração de um sistema em seus diversos níveis, os testes sejam feitos contra os próprios requisitos do componente/interface que está sendo testado(a), em contraste com modelos anteriores onde o componente era testado contra a especificação do componente/interface. Notar a diferença entre requisito e especificação.

O Modelo V virou um padrão da indústria de software depois de 1980 e, após o surgimento da Engenharia de Sistemas, tornou-se um conceito padrão em todos os domínios da indústria. O mundo do software tinha feito poucos avanços em termos de maturidade, em achar na bibliografia corrente as referências que poderiam se aplicar ao sistema.



D- Qual a diferença entre C e C++?

Quem sabe programar em C++, é capaz de programar C, devido à semelhança entre as linguagens e o fato do C++ ser uma extensão do C. Contudo o C não é completamente um subconjunto do C++. Grande parte de código C pode ser perfeitamente compilado em C++, mas existem algumas pequenas diferenças sintáticas e semânticas entre as linguagens que tornam alguns trechos de código C válidos em código C++ inválido, ou códigos que exibem comportamentos diferentes em cada linguagem. Algumas diferenças básicas:

O C permite a conversão implícita entre o tipo de dado `void*` para ponteiros para outros tipos, algo que o C++ não permite.

O C permite que constantes de caracteres sejam inseridas em chamadas de funções com parâmetros tipo `char*`, em C++ é preciso declarar o parâmetro como `const char`.

Além destas pequenas diferenças, C++ tem um conjunto de características que a torna fundamentalmente diferente de "C". Esse conjunto torna possível programar em C++ de um modo totalmente diferente do modo de programar da linguagem "C". O que traz a diferença é o modo da orientação na montagem do código.

Chamamos o modo de programar em "C" de orientado a procedimentos e chamamos o modo do "C++" de orientado a objetos. Muitas pessoas confundem as coisas quando começam a programar usando um compilador C++, pois esta linguagem permite programar nos dois modos. Essa é uma das características que a torna mais flexível.

Apesar de C++ permitir programar em modo orientado a procedimentos, podemos dizer que nestes casos estamos programando em "C", usando um compilador C++. Quando usamos C++ programamos em modo orientado a objetos. Devido a estas características, o C++ permite programar em modo misto, ou seja, escrevendo partes do código orientadas a procedimentos e outras orientadas a objetos.

As diferenças entre os dois modos de programar serão esclarecidas nos capítulos subsequentes. Por hora nos basta deixar claro que os dois modos são diferentes. Usar estes dois modos de programar ao mesmo tempo é uma das facilidades que o C++ permite, enquanto que outras linguagens orientadas a objetos como Java, Eiffel, etc, não permitem.