

1) Explique, com suas próprias palavras e/ou ilustrações, quais são os passos que o desenvolvedor C precisa realizar para produzir o arquivo executável final da aplicação. Sua resposta deve mostrar de forma clara:

1.1 - Onde ocorre a criação do código fonte e a diferença entre processos de compilação e “linkagem”;

1.2 - Quais são os arquivos gerados em cada passo do processo;

1.3 - Informar se o arquivo executável final pode funcionar em várias plataformas (Windows, GNU/Linux, e.g.) e o porquê;

1.4 - Comparar com o processo utilizado pela linguagem Python

RESPOSTA: O código-fonte, pode ser feito através de um editor de texto ou IDE, a linkagem é o processo que une o código ao sistema operacional, arquivos e as bibliotecas, para a execução de um programa, o código-fonte tem de, primeiramente, passar por um compilador que gera um código-objeto preparado para que a máquina o execute, por isso o executável não roda em outro sistema operacional, porque o código-objeto é esquematizado para o sistema operacional em que ele foi desenvolvido. Nesse caso, erros de sintaxe de linguagem aparecem no próprio compilador. Já os códigos-fonte de linguagens interpretadas são executados na máquina diretamente, por meio de outro programa chamado interpretador. Nisso os erros de sintaxe são identificados apenas durante a execução de programas.

Geralmente, as linguagens compiladas precisam de um procedimento de criação de um código-objeto de baixo nível (binário), enquanto os programas interpretados são analisados no momento de sua execução pelos chamados interpretadores. Quando o GCC é rodado, ele realiza quatro etapas para gerar o executável: pré-processamento, compilação, montagem e linkagem, sempre nesta ordem. Já o interpretador faz esta 'tradução' em tempo real para código de máquina, ou seja, em tempo de execução. Já o compilador traduz o programa inteiro em código de máquina de uma só vez e então o executa, criando um arquivo que pode ser rodado (executável). O compilador gera um relatório de erros (caso eles existam) e o interpretador interrompe a tradução quando encontra um primeiro erro.

Em geral, o tempo de execução de um código compilado é menor que um interpretado já que o compilado é inteiramente traduzido antes de sua execução. Enquanto o interpretado é traduzido instrução por instrução, Python é um exemplo de linguagem interpretada.