

# Resumo: Estudo da Linguagem C

## Sequencia de Escape:

<code>\n</code>	Nova linha. Posiciona o cursor no início da nova linha.
<code>\t</code>	Tabulação horizontal. Move o cursor para a próxima marca de tabulação.
<code>\r</code>	Carriage return (CR). Posiciona o cursor no início da linha atual.
<code>\a</code>	Alerta. Faz soar a campainha do sistema.
<code>\\</code>	Imprime o caractere de barra invertida (backslash) em uma instrução printf.
<code>\"</code>	Imprime o caractere de aspas duplas em uma instrução printf.

## Dicas retiradas do livro: Como Programar em C, de Paul J. Deitel e Harvey M. Deitel, 6ª Edição

## Dicas de Portabilidade:

1. Como o C é uma linguagem independente de hardware e amplamente disponível, as aplicações escritas em C podem ser executadas com pouca ou nenhuma modificação em uma grande variedade de sistemas computacionais.
2. Usar as funções da biblioteca padrão do C em vez de escrever suas próprias versões similares pode melhorar a portabilidade do programa porque essas funções estão colocadas em praticamente todas as implementações do ANSI C.
3. Embora seja possível escrever programas portáteis, há muitos problemas entre as diferentes implementações do C e os diferentes computadores que tornam a portabilidade um objetivo difícil de atingir. Simplesmente escrever programas em C não garante a portabilidade.

## Dicas de Desempenho:

1. Usar as funções da biblioteca standard do C em vez de você escrever suas próprias versões similares por melhorar o desempenho do programa porque essas funções foram desenvolvidas cuidadosamente por pessoal eficiente.

## Boas Práticas de Programação:

1. Escreva seus programas em C de uma maneira simples e objetiva. Algumas vezes isto é chamado KIS (“keep it simple”), mantenha a simplicidade. Não complique a linguagem tentando soluções estranhas.
2. Leias os manuais da versão do C que estiver usando. Consulte frequentemente estes manuais para se certificar do conhecimento do rico conjunto de recursos do C e de que eles estão sendo usados corretamente.
3. Seu computador e compilador são bons mestres. Se você não estiver certo de como funciona um recurso em C, escreva um programa de teste que utilize aquele recurso, compile e execute o programa, e veja o que acontece.
4. Todas as funções devem ser precedidas por um comentário descrevendo seu objetivo.
5. O último caractere impresso por uma função que realiza qualquer impressão deve ser o de nova linha (`\n`). Isto assegura que a função deixará o cursor da tela posicionado no início de uma nova linha. Procedimentos desta natureza estimulam a reutilização do software – um objetivo principal em ambientes de desenvolvimento de software.
6. Faça o recuo de um nível (três espaços) em todo o texto (corpo) de cada função dentro das chaves que a definem. Isto ressalta a estrutura funcional dos programas e ajuda a torná-los mais fáceis de ler.
7. Determine uma convenção para o tamanho de recuo preferido e então aplique-a uniformemente. A tecla de tabulação podem variar. Recomendamos usar paradas de tabulação de  $\frac{1}{4}$  da polegada (aproximadamente 6 mm) ou recuar três espaços para cada nível de recuo.

## Erro de Programação:

1. Esquecer de encerrar um comentário com `*/`.
2. Começar um comentário com os caracteres `*/` ou terminar com `/*`.
3. Em um programa, digitar como `print` o nome da função de saída `printf`.