

1. Como funcionam os comandos de laço controlados por contador?
Define o parâmetro de início, de fim e o passo de incremento. Testa o parâmetro de início uma única vez e outros uma vez por incremento, testando a condição sempre antes de entrar no laço. ✓
2. Por que os comandos de iteração são chamados de “poder da computação”?
Antes deles os programadores teriam de informar cada passo, criando programas maiores e não genéricos. ✓
3. Como um comando **switch** pode ser simulado nas linguagens que não dão suporte para condicionais múltiplas?
Utilizando condicionais de dois caminhos aninhados. ✓
4. Explique a afirmação: os comandos condicionais múltiplos podem afetar a confiabilidade e a legibilidade dos programas.
Caso um condicional múltiplo não tenha um argumento de parada, 'break', ao entrar em um dos casos, ele vai entrar em todos os casos até encontrar um 'break', ou terminar o caso padrão. ✓
5. Em várias linguagens a regra semântica é: o **else** é do **if** mais próximo que ainda não tem **else**. Como é possível forçar uma semântica alternativa?
Modifique o seguinte trecho de código para que ele realize a semântica sugerida pela sua disposição textual.

```
if (x == 7)
    if (y == 11)
        z = 13;
    else z = 17;
```

```
if (x == 7) {
    if (y == 11) {
        z = 13;
    }
    else {
        z = 17;
    }
}
```

✓

6. Apresente vantagens e desvantagens de fornecer apenas variáveis locais estáticas em subprogramas.
V: Maior facilidade ao alocar memória e sensibilidade ao histórico.
D: Não suportar recursão e não compartilhar armazenamento com variáveis locais de subprogramas inativos. ✓

7. As variáveis globais podem ser visíveis dentro dos subprogramas em várias linguagens de programação. Qual a restrição associada imposta pela maioria das linguagens com relação a esse fato? Qual o motivo dessa restrição?
Manipular seu endereço de memória. Evitar que o programador mude-a indevidamente. ✓
8. Explique os termos relacionados aos subprogramas: parâmetros formais, parâmetros reais, protocolo e perfil.
Perfil: número, ordem e tipo dos parâmetros formais. ✓
Protocolo: tipo de retorno mais o perfil. ✓
Parâmetro real: tipo de retorno. ✗
Parâmetro formal: número, ordem e tipo na declaração do parâmetro. ✓
9. Discuta algumas questões de projeto relacionadas aos subprogramas.
Alocação estática ou dinâmica; ✓
Aninhamento de subprogramas ou não; ✓
Acesso aos dados de forma direta ou por parâmetro.
10. Por que subprogramas aninhados não são implementados na maioria das linguagens de programação?
Os projetistas acreditam ter formas melhores de organizar os programas e sua complexidade para permiti-los e menos vantajoso do que permiti-los. ✓