

**Roteiro de
Estudos**



**Algoritmos e Técnicas de
Programação**

Este roteiro orientará a sua aprendizagem por meio da leitura de livros e artigos que cabem na sua rotina de estudos. Experimente esse recurso e aumente a sua habilidade de relacionar a teoria à prática profissional.

No seu caminho de aprendizagem, você encontrará os seguintes tópicos:

- ✓ Texto de apresentação de cada leitura indicada;
- ✓ Links para acesso às referências bibliográficas.

É importante ressaltar: o seu esforço individual é fundamental para a sua aprendizagem, mas você não estará sozinho nessa!

UNIDADE 3

Fundamentos de algoritmos e das linguagens de programação

Estruturas de decisão condicional

Caro aluno, no dia a dia, todas as ações que realizamos nos deparamos com alguma condição para que uma ação possa ou não ser realizada, observe que a lógica faz parte no nosso modo de pensar e resolver as coisas. Por exemplo, quando você vai encher um copo de água, abre a torneira, se a água estiver próximo a transbordar, você fecha a torneira. Você teve que decidir quando fechar, esse processo chamamos de decisão condicional.

Segundo Manzano (2013), para a solução de um problema envolvendo situações, podemos utilizar a instrução “if”, em português “se”, cuja função é tomar uma decisão e criar um desvio dentro do programa, para que possamos chegar a uma condição verdadeira ou falsa. Lembrando que a instrução pode receber valores em ambos os casos.

Toda linguagem de programação tem sua estrutura de programação, na Linguagem C, o condicional segue o formato:

```
if <(condição)>
{
    <conjunto de comandos>;
}
```

Esse modelo é chamado de condicional simples, pois, somente se ocorrer o fato o resultado será realizado, caso contrário, o fluxo do programa segue normal. Por exemplo, você está escrevendo o seu TCC, a condição de parar é se o telefone tocar, se não acontecer continuará escrevendo. A instrução if, significa SE, então na linguagem ficaria assim: `if (telefone == tocar) {atendo}`.

Existe a condição composta, que permite que outra situação seja realizada, por exemplo, se estiver chovendo ficará em casa, senão, vai jogar bola. Na linguagem C, aparece o comando else, que significa senão. Se fossemos programar a situação exemplo, ficaria assim:

```
Se (tempo == chovendo) {vou ficar em casa}
```

```
Senão {vou jogar bola}
```

Observe que é grande a semelhança com analisamos as nossas situações do cotidiano, mas ainda existe outro método condicional que permite analisar várias situações, que denominamos estrutura condicional de seleção de casos, segundo Schildt (1997, p. 35), “testa sucessivamente o valor de uma expressão contra uma lista de constantes inteiras ou de caractere”. Quando os valores são avaliados o comando é executado.

Se houver a necessidade de múltiplas comparações, existe o comando switch, imagine seu sistema com um menu com várias opções, uma forma de reduzir a complexidade de vários if é o uso do comando switch.

Por exemplo, você tem um determinado orçamento, está no shopping, irá passar por todas as lojas (opções do sistema) mas, somente quando a oferta estiver dentro das suas possibilidades a compra (processamento) será realizada.

Referências e Link do material na Biblioteca Virtual e artigo

MANZANO, J. A. N. G. **Algoritmos: técnicas de programação**. 2. ed. São Paulo: Érica, 2015.

SCHEFFER, Vanessa Cadan. ARTERO, Marcio Aparecido. **Algoritmos e técnicas de programação** – Londrina: Editora e Distribuidora Educacional S.A., 2018.

SCHILDT, H. **C Completo e total**. 3. Ed. São Paulo: Pearson Prentice Hall, 2005. SOFFNER, R. **Algoritmos e Programação em Linguagem C**. 1. ed. São Paulo: Saraiva, 2013.

Estruturas de repetição condicional

Salve estudante, muitos processamentos tem a necessidade em repetir alguns processos, as linguagens oferecem recursos para isso, normalmente chamados de loop ou laços de repetição.

Um dos comandos é o while ou do-while, significa enquanto, sempre associado a uma condição que irá determinar se o laço continua ou não. Por exemplo, um robô deve tampar algumas garrafas, chegou um lote e não se sabe a quantidade de garrafas. Como o robô irá interpretar o processo assim, enquanto houver garrafa o processo de tampar continuar, quando não houver mais, parar.

Segundo Soffner (2013), quando aplicado o comando do-while, as ações poderão ser executadas pelo menos uma vez antes do teste condicional. Nesse caso, é possível ter uma sequência de repetição de processos, sendo assim, é possível ter várias opções para a solução de um problema em um programa.

Exemplificando em código, ficaria assim:

```
1. int contador = 1; //declarando e inicializando a variável de controle
2. while (contador <= 10) // Testando a condição
3. {
4.     printf("%d ", contador); //Executando um comando dentro do laço
5.     contador++; //atualizando a variável de controle
6. }
7. return 0;
```

Os comandos entre chaves serão executados até que o contador assuma um valor maior que 10, isso irá cancelar o loop e a linha 7 (return 0). Por exemplo, uma máquina recebe um lote de mercadoria para inserir um rótulo, cada lote tem uma quantidade diferente, como proceder? Na programação estará inserido, enquanto houver material, proceder com os rótulos, quando não houver mais, parar o processamento. Observe como é intuitivo, idêntico ao pensamento humano, enquanto tiver mercadoria vou rotulando.

Referências e Link do material na Biblioteca Virtual e artigo

SCHEFFER, Vanessa Cadan. ARTERO, Marcio Aparecido. **Algoritmos e técnicas de programação** – Londrina: Editora e Distribuidora Educacional S.A., 2018.

SOFFNER, R. **Algoritmos e Programação em Linguagem C** . 1. ed. São Paulo: Saraiva, 2013.

Estruturas de repetição determinísticas

Caro aluno, todas as linguagens oferecem comandos para elaboração do loop ou laço de repetição, o comando while, geralmente é associado quando não se sabe a quantidade de itens que serão processados, quando o valor de processo é conhecido, o comando FOR é utilizado. Não existe uma regra ou obrigatoriedade no uso dos comandos, só a praticidade que cada comando possui.

Sua sintaxe é: For (variável de inicialização; condição; incremento), onde:

Variável de inicialização: Atribuição que inicia uma variável que faz o controle do laço.

Condição: expressão usada que determinará o final do laço.

Incremento: define a variável de controle e muda a cada passada no laço.

Por exemplo, um lava jato recebe 10 carros, você irá programar o seu sistema para processar somente os 10 automóveis, no seu programa teria o seguinte código: for (carro=0; carro<10; carro++).

Nada é estável dentro de uma programação, se houver a necessidade de sair do laço sem terminar a contagem, isso é possível com o comando break. Por exemplo, na linha de produção o seu robô foi programado para pintar 200 portas, um sensor avalia a qualidade, aparece uma porta defeituosa, o comando break é executado, o laço é quebrado e outra ação fora do laço será realizada. Note que na programação o programador tem total domínio do código, por isso o estudo da lógica, todas as possibilidades devem ser registradas.

Dentro do laço FOR é possível pular um ou mais itens, existe um comando de desvio incondicional o continue, dentro de um laço faz com que a execução da instrução atual seja interrompida e passa para a próxima iteração do laço.

Segundo Damas (2016), uma instrução continue dentro de um laço possibilita que a execução de comandos corrente seja terminada, passando à próxima iteração do laço, ou seja, quando usamos o continue dentro de um laço, passa o laço para a próxima iteração.

Hoje em dia, as empresas estão utilizando robôs para quase todas as operações de fabricações de produtos, os movimentos são repetitivos, então, veja o quanto é importante os loops dentro

dos programas, um erro e todo projeto pode vir a fracassar. Por isso, o profissional deve estar sempre atualizado e a lógica como principal bússola na programação.

Referências e Link do material na Biblioteca Virtual e artigo

SCHEFFER, Vanessa Cadan. ARTERO, Marcio Aparecido. **Algoritmos e técnicas de programação** – Londrina: Editora e Distribuidora Educacional S.A., 2018.

DAMAS, L. **Linguagem C**. 10. ed. Rio de Janeiro: LTC, 2016.