

**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 1

Fundamentos de algoritmos e das linguagens de programação

Introdução aos algoritmos

Prezado aluno, todo pensamento tem em vista tentar resolver um problema, seja do mais simples ao mais complexo, desde abrir uma porta, até encontrar a equação do orçamento de sua empresa. Para que esse raciocínio tenha uma resposta, todas as análises devem ser colocadas em ordem lógica. Antes de dar continuidade, vale saber que um algoritmo é uma sequência finita de passos que podem levar à criação e execução de uma determinada tarefa com a intenção de resolver uma problemática FORBELLONE E EBERSPÄCHER (2005).

Todos os dias, realizamos ações que tem início, meio e fim, para que isso aconteça é necessário o que denominamos lógica. Observe que nascemos com essa capacidade de análise da situação com o objetivo de resolver um problema. Segundo Manzano (2015), um algoritmo é um conjunto lógico de operações predefinidas que resolva um determinado problema de forma intuitiva. Apesar de todos os avanços tecnológicos, continuamos com o mesmo processo, analisar, obter entendimento em ordem lógica e resolver ou tentar resolver, se não fosse isso, nem teríamos esse avanço tecnológico.

Veja como é fácil, existe um problema, isso irá gerar pistas, dados e o pressuposto retorno, então, é feita a análise para que a resposta aconteça de forma eficiente. Nesse momento, acontece o algoritmo, não existe ainda nenhuma linguagem de programação, apenas o entendimento do problema e possíveis soluções.

A partir do algoritmo realizado, o programador pode transcrever para uma linguagem, ou no caso do dia a dia, resolver o problema. O que foi feito? Apenas pensamos e geramos uma solução para um problema qualquer, no caso de ser um programa, uma linguagem de programação foi necessária.

Outra técnica muito utilizada para documentar a lógica (ideias) são os pseudocódigos é uma ferramenta de programação que permite aos programadores de computador para desenvolver a lógica antes de executar um código particular. É uma forma entre a linguagem de programação e a linguagem natural

A linguagem natural tem uma grande contribuição quando vamos desenvolver uma aplicação computacional, pois pode direcionar de forma simples e eficiente as descrições dos problemas e suas soluções.

Estamos falando em registrar o pensamento humano em registros que possam ser divulgados, uma ferramenta muito útil, são os fluxogramas que são diagramas que descrevem um sistema ou algoritmo de computador. Podem ser gráficos simples e desenhados à mão ou diagramas abrangentes desenhados por computador descrevendo as várias etapas e rotas. Devido a sua fácil visualização e entendimento, são amplamente utilizados para comunicar e melhorar complexos processos através de diagramas. Sua linguagem são formas geométricas como retângulos, paralelogramos, losangos e muitas outras formas para definir os tipos de ações, bem como setas conectoras que irão definir fluxo e sequência.

O pseudocódigo é aplicado para ter uma ideia mais próxima sobre o potencial do programa que está realizando, a facilidade desse processo, possibilita executar o programa e verificar como o resultado será visualizado. Com o pseudocódigo, o programador tem uma visão mais próxima de como será o código-fonte com uma linguagem de programação.

Referências e Link do material na Biblioteca Virtual e artigo

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F. **Lógica de programação: a construção de Algoritmos e Estruturas de Dados**. São Paulo: Makron, 2000.

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

SANTOS, D. **Processamento da linguagem natural: uma apresentação através das aplicações**. Ranchhod (org.). Lisboa: Caminho, 2001.

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

Conceitos Básicos de Linguagens de Programação

Caro aluno, todos os seres do planeta para que possam conviver, precisam se comunicar de forma verbal, visual, auditiva, etc. No caso dos computadores, isso não é diferente, as linguagens computacionais foram elaboradas de forma que o programador (pessoa) possa interagir com a máquina, lembrando que existe a forma lógica de comunicação (algoritmo).

O algoritmo é uma técnica onde o programador pode escrever as ideias, analisar e posteriormente escrever em uma linguagem de programação para que o computador possa realizar os processos. Existe uma grande diversidade de linguagens que irão se adequar a necessidade do problema a ser resolvido, por exemplo, linguagem de máquina, Linguagem

Assembly, Pascal, C, Java, PHP, Ruby dentre outras. Não se assuste, encare como vários países e cada um com seu idioma próprio

Para Tucker (2010), da mesma forma que entendemos as linguagens naturais, essas utilizadas por todos no dia a dia, a linguagem de programação é a comunicação de ideias entre o computador e as pessoas.

Segundo Marçula (2013, p. 170), a linguagem de programação pode ser entendida como um conjunto de palavras (vocabulário) e um conjunto de regras gramaticais (para relacionar essas palavras) usados para instruir o sistema de computação a realizar tarefas específicas e com isso, criar os programas. Cada linguagem tem o seu conjunto de palavras-chave e sintaxes.

Dentre as linguagens, podemos destacar linguagens de baixo nível, chamadas assim por serem escritas em linguagem de máquina, os códigos não são próximos as linguagens naturais, aparecem código assim: *mov AL, [SI], cmp AL, 0, je pfend*, e as linguagem de alto nível, assim por se assemelharem com as linguagens naturais.

As linguagens de alto nível são facilitadoras para os programadores, mas para os computadores não, as máquinas entendem apenas linguagem de máquina, então, foram criados alguns métodos de tradução denominados: interpretador, compilador e tradutor. Onde podemos conceituar que:

O interpretador traduz em tempo real, a linguagem de alto nível é convertida em um código intermediário (ainda não é código de máquina), então, quando o programa é chamado, o interpretador converte em linguagem de máquina e executa, por exemplos a linguagem Basic, Prolog, Python e Java.

O compilador traduz o código inteiro e gerar um arquivo executável, por exemplo, Delphi, Cobol, C++, C#, C.

O tradutor, realiza uma economia de memória, ao invés de armazenar o programa fonte, transforma-o em código intermediário, que não consome muita memória, em seguida são transformadas em linguagem de máquina e executadas.

A área de programação gera oportunidades em várias áreas de programação, gerando oportunidades em empregos como: Administrador de banco de dados, analista de redes, programador mobile, programador web, atuar no setor de TI de uma empresa para desenvolvimento de aplicações.

Referências e Link do material na Biblioteca Virtual e artigo

MARÇULA, M. **Informática: conceitos e aplicações**. 4. ed. rev. São Paulo: Érica, 2013.

TUCKER, A. B. **Linguagens de programação: princípios e paradigmas**. Porto Alegre: AMGH, 2010.

Componentes e elementos de Linguagem de Programação

O raciocínio lógico, é o entendimento do problema seguido da análise para resolução, as documentações dessas ideias podem ser realizadas em forma de linguagem natural, algoritmos, fluxogramas e pseudocódigo. Até esse ponto, o programador consegue visualizar o que será programado, mas o computador tem suas próprias linguagens. Cada linguagem tem suas próprias características.

Segundo Manzano (2015), os algoritmos criados por você deverão ser convertidos em linguagens de alto nível, como, por exemplo: a linguagem Java, Pascal, C, C++ entre outras conceituadas nesta seção. Você deverá escrever os códigos fontes na linguagem escolhida para que os mesmos se tornem executáveis, porém, cada linguagem adota um método particular para gerar o código executável.

Nas linguagens de programação, existem inúmeros elementos específicos de cada linguagem, cabe ao programador escolher a que melhor se adapta ao problema e modelar o programa.

Para todos os tipos de linguagem de programação é necessário montar seu algoritmo, isto é, estabelecer a estrutura dessa linguagem, sendo esse um processo comum a todos os tipos. Criar variáveis, definir os seus tipos: inteira, caractere, decimal, ou se é uma constante, verificar o tipo de processamento que será realizado e como os resultados serão apresentados.

A linguagem C é muito importante para a história da computação, o sistema operacional do seu computador foi escrito em linguagem C, muitos drives e o Linux foi totalmente foram escritos em linguagem C, então, a importância em conhecer a linguagem C.

A linguagem C possui uma série de funções com recursos adicionais e disponíveis para utilização, essas funções são chamadas de bibliotecas, que o programador pode inserir em acordo da necessidade do programa, assim o código final fica enxuto. Existe uma diversidade de bibliotecas como por exemplo: `stdlib.h`, `stdio.h`, `math.h`, `string.h` e, se houver necessidade o programador pode criar suas próprias bibliotecas.

Segundo Spolsky, A linguagem C nos permite fazer diferentes aplicações, escritas em diferentes linguagens, interoperarem entre si. A grande maioria das linguagens de programação proveem suporte a programação de interfaces em C.

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.

Joel Spolsky, **“Advice for Computer Science College Students”**, Joel on Software, Janeiro 2005. www.joelonsoftware.com/articles/CollegeAdvice.html.