

Apresentação

Da mesma forma que existem vários idiomas no mundo, na computação existem várias linguagens de programação, as LPs. LP é uma linguagem na qual os programas são escritos e que faz uma tradução do algoritmo para uma linguagem que o computador entenda.

Nesta Unidade de Aprendizagem, você vai conhecer a linguagem C, também conhecida como a linguagem das linguagens, pois é bastante difundida no meio acadêmico e usada para que você dê os primeiros passos no mundo da programação. Você vai entender como surgiu a linguagem por meio do histórico da linguagem C, compreender a estrutura básica da linguagem, conhecer vários ambientes de desenvolvimento e configurar o que melhor se adaptar às suas necessidades.

Bons estudos.

Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:

- Reconhecer o histórico da linguagem C.
- Identificar a estrutura básica de um programa em C.
- Instalar ambientes de desenvolvimento da linguagem C.

Videoaula



Vídeo



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Desafio

Você acabou de se formar no curso de Engenharia de Computação e recebeu o convite de mais dois colegas que também fizeram o curso de computação, porém em outras universidades e com foco diferente do seu, para iniciarem uma **startup de desenvolvimento de *software***.



Como você estudou a linguagem de programação C durante todo o curso, esse será o enfoque do seu estudo.

Assim, você precisa escolher um ambiente de desenvolvimento para a linguagem C e realizar a sua instalação. Você deve fazer uma pequena introdução do ambiente de desenvolvimento que foi escolhido, indicar qual sistema operacional utilizou e o passo a passo para a instalação, finalizando com um *print* da IDE.

Infográfico

A linguagem de programação C é uma linguagem compilada: é preciso que o código-fonte passe por um programa chamado *compilador*. A saída desse processo é uma espécie de codificação que o computador entende.

Acompanhe no infográfico como esse processo acontece.

CRIANDO UM PROGRAMA EM LINGUAGEM C



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.



```
1 #include <stdio.h>
2 int main()
3 {
4     printf("Bem-vindo ao seu primeiro Jogo em C!\n");
5     return 0;
6 }
```

COMPILADOR C

Esse é o arquivo que contém o **código-fonte**; nele, estão os comandos do seu programa. Você pode criá-lo usando um editor de texto comum ou uma IDE.

MEU PRIMEIRO PROGRAMA

Para compilar um programa usando o compilador, **você precisa digitar o seguinte comando no terminal:**
gcc -o meuprimeiroprograma.c
meuprimeiroprograma

No final, você terá um programa executável!



Conteúdo do Livro

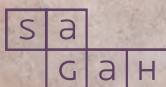
Para desenvolver programas, é preciso conhecimento de alguma linguagem de programação. Neste capítulo, você aprenderá um pouco mais sobre o que são essas linguagens e como elas funcionam, e entrará em contato com a linguagem C, uma das mais populares.

Leia o capítulo A linguagem C - Conceitos básicos, da obra *Algoritmos de Programação*, que serve de base teórica para esta Unidade de Aprendizagem.

Boa leitura.

ALGORÍTMOS DE PROGRAMAÇÃO

Marcela Santos



SOLUÇÕES
EDUCACIONAIS
INTEGRADAS



A linguagem C – conceitos básicos

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Conhecer o histórico da linguagem C.
- Identificar a estrutura básica de um programa em C.
- Instalar e conhecer ambientes de desenvolvimento da linguagem C.

Introdução

Da mesma forma que existem vários idiomas no mundo, na computação, existem várias linguagens de programação, as LPs. Uma LP é uma linguagem na qual os programas são escritos e que faz uma tradução do algoritmo para uma linguagem que o computador entenda.

Neste texto, você vai estudar sobre a linguagem C, também conhecida como a “linguagem das linguagens”, por ser bastante difundida no meio acadêmico e usada para que você dê os primeiros passos no mundo da programação. Você vai entender como surgiu a linguagem por meio do histórico da linguagem C, compreender a estrutura básica da linguagem, além de conhecer vários ambientes de desenvolvimento e configurar o que melhor adaptar-se às suas necessidades.

Histórico da linguagem C

Antes de começarmos a falar sobre C, que é a linguagem que usaremos, vamos entender o que é uma linguagem de programação. No capítulo **Introdução à Lógica de Programação**, falamos sobre Algoritmos e como o seu papel é importante no desenvolvimento de um programa. A Figura 1, a seguir, faz a representação de como um algoritmo ajuda na resolução de problemas.

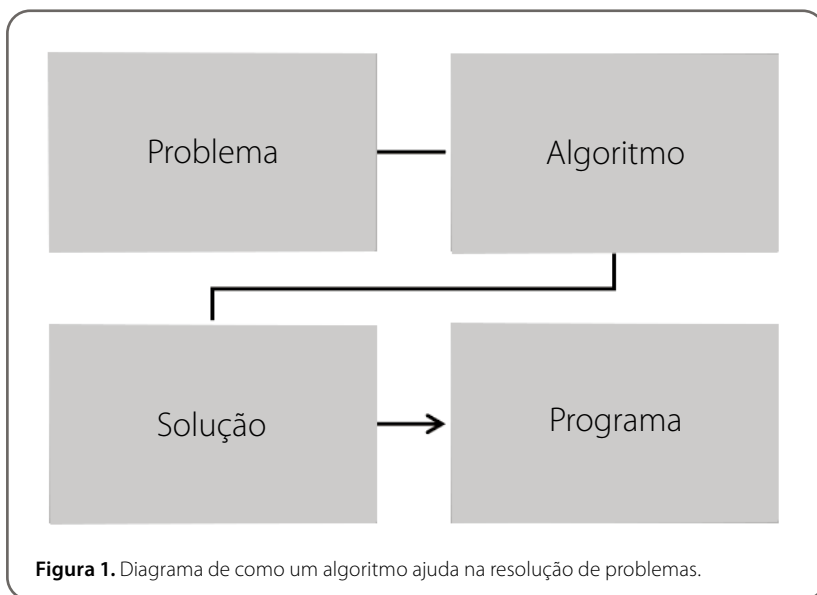


Figura 1. Diagrama de como um algoritmo ajuda na resolução de problemas.

Porém, se o algoritmo não está escrito em uma linguagem que o computador entenda, é preciso, então, fazer uma tradução da linguagem que nós, humanos, usamos para a que a máquina entenda. Essa tradução é feita por meio da utilização de uma linguagem de programação (LP). Existem várias linguagens de programação, com paradigmas e níveis de abstração diferentes.

Quando uma LP assemelha-se à linguagem falada ou escrita por um ser humano, ela é dita de alto nível, e, quando ela aproxima-se da linguagem de máquina, é dita de baixo nível. Da mesma forma que o Português, por exemplo, tem regras (sintaxe) para que seja compreendido por qualquer pessoa que fale esta língua, as LPs também têm um conjunto de regras com as quais é possível traduzir um algoritmo e gerar um programa.

Dito isso, podemos começar a falar da linguagem que vamos utilizar neste livro: a linguagem de programação C. A linguagem C é uma LP bastante popular, criada por volta de 1970, por Dennis Ritchie.

A linguagem C foi desenvolvida nos laboratórios Bell, onde muitas de suas ideias iniciais surgiram de outra linguagem, a B, que tem como antecessoras as linguagens BCPL e CPL. CPL foi uma linguagem desenvolvida com o propósito de ser uma linguagem de alto e baixo níveis. A maior desvantagem da CPL foi que era muito pesada para o desenvolvimento de algumas aplicações. Em 1967, foi desenvolvida a Basic CPL – BCPL, que mantinha as características básicas da CPL. Ken Thompson, que trabalhava na Bell Labs, utilizou esse projeto para

o desenvolvimento da linguagem B. Assim, B foi uma versão de BCPL escrita para o desenvolvimento de sistemas. Em 1972, Dennis Ritchie, trabalhando com Ken Thompson, desenvolveu uma linguagem também baseada na linguagem BCPL, mas que mantinha parte da especificação para acesso a hardware, ou seja, a parte das características de baixo nível da linguagem, conhecida como C.

A linguagem C é usada por programadores há muito tempo e poderosa no desenvolvimento dos mais diversos tipos de sistema, desde sistemas operacionais, passando por compiladores, editores de texto, etc. Além da diversidade de sistemas, é importante ressaltar que C é uma linguagem que pode ser utilizada em máquinas com um alto poder de processamento, bem como em máquinas mais simples e com baixo poder de processamento.

Dentre as características de C, podemos destacar:

- Portabilidade: os programas em C são compilados, gerando um único executável. Assim, é possível compilar um programa escrito em C para qualquer máquina, desde que exista compilador C para ela. Na prática, existem compiladores C para quase todos os computadores.
- Poder e variedade dos operadores: é possível trabalhar com funções matemáticas, criação e manipulação de arquivos de Dos, dentre outras possibilidades. Isso tudo pode ser feito de forma simplificada, com adição de bibliotecas padronizadas, presentes na própria linguagem.
- Sintaxe elegante, estruturada e flexível: indicada para quem está começando a programar.
- Acesso facilitado à memória e a todo o hardware, quando preciso: assim, é possível ir de um programa simples até o desenvolvimento de um sistema operacional.
- Uso de procedimentos e funções para desenvolver sistemas desacoplados: sistemas desacoplados são sistemas onde é possível isolar pequenas partes para, por exemplo, encontrar erros sem causar nenhum dano ao restante do sistema.

Pela natureza de como foi desenvolvida, C também é chamada de linguagem de Médio Nível, pois possui um conjunto de instruções necessárias para acesso ao hardware, como as linguagens de baixo nível (*Assembly*). Além disso, C pode ser simples e amigável, como uma linguagem de alto-nível deve ser. Uma linguagem de médio nível pode, em alguns casos, não prover as estruturas necessárias a uma linguagem de alto nível, mas certamente ela tem ferramentas que podem ser usadas como blocos para construção dessas estruturas necessárias.



Saiba mais

A primeira versão de C foi criada por Dennis Ritchie, em 1972, nos laboratórios Bell, para ser incluída como um dos softwares a serem distribuídos juntamente com o sistema operacional Unix do computador PDP-11.

Estrutura básica da linguagem C

Neste ponto, você deve estar querendo saber como é escrito um programa que utiliza a linguagem C. A primeira coisa que temos que entender é o processo da criação desse programa.

- Com o algoritmo em mente, você deve traduzi-lo para C em um arquivo e salvá-lo com a extensão C. Por exemplo, imagine que eu escrevi meu primeiro programa em um arquivo que irei nomear de `programa1`; ao salvá-lo, esse arquivo precisa de um nome e de uma extensão; assim, esse arquivo ficaria como: `programa1.c`
- A segunda parte é usar um compilador para gerar o programa. O compilador nada mais é do que um programa que vai fazer toda a conversa e tradução do seu código-fonte para a linguagem mais baixa do sistema, a linguagem da máquina.
- Pronto, agora seu programa foi criado. Basta somente executá-lo.
- A Figura 2, a seguir, demonstra, por meio de um esquema, a criação de um programa compilado.

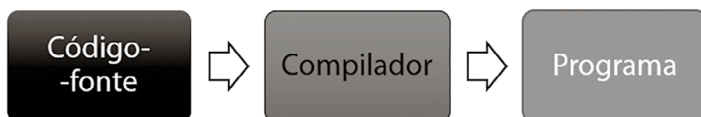


Figura 2. Como um programa compilado é criado?

Agora vamos entender a estrutura básica do nosso código-fonte, escrito em linguagem C. Mais à frente, veremos com o uso das ferramentas precisamos, mas, por hora, imagine que você abriu qualquer editor de texto da sua preferência e criou um arquivo novo, salvou-o com extensão C e, agora, chegou a hora de preenchê-lo com seu código em C, conforme exemplificado na Figura 3.

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("Bem-vindo ao seu primeiro Jogo em C!\n");
5     return 0;
6 }
```

Figura 3. Primeiro programa em C.

Durante todo este livro, vamos explicar os códigos, usando os números das linhas na margem à esquerda dos códigos. Vamos a elas:

- Falamos sobre uma das características do C, o uso de bibliotecas padronizadas. A linha 1 é o comando para adicionar a biblioteca padrão `stdio.h` no seu código-fonte. Uma das possibilidades que ela traz são comandos de entrada e saída de dados.
- Na linha 2, temos a declaração da função principal, que, em C, é chamada de `main`, e a palavra `int` antes dela indica que deverá “devolver” algo do tipo inteiro. Depois de `main`, temos um par de parênteses vazio, indicando que ela não tem nenhum argumento.
- Nas linhas 3 e 6, temos um par de chaves, entre as quais a magia acontece, onde todo o seu algoritmo, agora traduzido para C, ficará.
- Na linha 4, temos um exemplo de instrução que escreve um texto na tela.
- Por fim, na linha 5, temos o valor que a função `main` retorna, 0.

Sabemos que, a essa altura do livro, poucas dessas palavras parecem familiares, mas calma: foi assim com todos os programadores profissionais — todo mundo, no início de aprendizado, passou por essa etapa. É preciso que neste início (principalmente) você estude e pratique como se estivesse aprendendo um novo idioma, como o inglês. Só assim tudo ficará mais familiar e, como os programadores dizem, “no sangue”.

No exemplo da Figura 4, o programa escreve algo na tela. Se você quiser escrever um programa, por exemplo, que soma 2 números, deverá substituir a linha 4 por toda a tradução do seu algoritmo. Então, a estrutura básica de um código C é a seguinte:

```
1 #include <stdio.h>
2 int main()
3 {
4     /*Aqui fica seu código. Apague essa linha e adicione o que
5     seu algoritmo traduzido irá fazer!!*/
6     return 0;
7 }
```

Figura 4. Estrutura básica de um programa em C.

Agora temos o código-fonte. Na próxima seção, vamos conhecer os ambientes para desenvolvimento e, assim, gerar o programa em C.

Ambientes de desenvolvimento com linguagem C

Existem dois caminhos a seguir em se tratando de desenvolvimento de programas. Você pode usar um editor de texto e um compilador/interpretador da linguagem que escolher, ou um ambiente integrado de desenvolvido, também conhecido como IDE.

Muitas pessoas dizem que, no início dos estudos de programação, é melhor usar um editor de texto, já que, em muitas vezes, uma IDE traz muita informação que ainda não é familiar ao estudante.

Neste livro, vamos dar a você sugestões de editores de texto e IDE, para que, assim, você possa fazer suas escolhas. Começando com os editores de texto:

- Sublime Text, que pode ser usado em ambientes OS X, Windows ou Linux.
- Atom, que também pode ser usado em OS X, Windows ou Linux.
- Editores-padrão do seu sistema operacional.

Agora, as IDEs:

- Dev-C++: ferramenta para desenvolvimento C e C++, para o ambiente Windows — essa ferramenta é bastante difundida no meio acadêmico.
- Code::Blocks: ferramenta open-source, na qual se pode programar em C, C++ e Fortran. (existe versão para Windows, Linux e OS X) — uma das ferramentas mais flexíveis e com uma quantidade grande de *plugins*, o que pode ajudar durante o desenvolvimento de algum programa específico.
- Visual Studio: ambiente de programação para o Windows e OS X — nessa IDE, é possível programar em várias linguagens, dentre elas a C.
- Eclipse: uma das IDEs mais completas, na qual se pode programar (realizando a configuração adequada) em diversas linguagens — disponível para OS X, Windows ou Linux.

Alguns detalhes:

- Se você escolher usar o editor de texto, é preciso instalar também o compilador C.
- Se você escolher usar uma IDE, deve configurá-la para a linguagem C, no caso das IDEs que possuem uma vasta gama de possibilidade de linguagens de programação.

Escolhido o ambiente de programação, vamos criar nosso primeiro programa? Vamos usar aqui o editor de texto+compilador no ambiente Linux. No caso das IDEs, é preciso somente abrir um novo documento, “codar” (termo da moda e moderninho pra programar!!!), e clicar no ícone responsável por compilar e/ou executar o programa.

Como o editor simples foi o escolhido, é preciso, agora, ir para a linha de comando do Linux e digitar alguns comandos. É necessário acessar a pasta onde o arquivo.c esteja e, em seguida, basta digitar o seguinte: gcc nomedoarquivo.c -o nomedoprograma.



Saiba mais

Linha de comando para compilação de um programa em C:

- gcc — comando para executar o compilador C;
- nomedoarquivo — o arquivo deve ter sido salvo com extensão C e é preciso estar na mesma pasta que ele;
- -o — opção necessária para gerar o executável em C;
- nomedoprograma — nome do programa que vai ser gerado; se não for digitado, o compilador gera um programa cujo nome será a.out.

Depois disso, e se tudo ocorreu bem, o compilador gera um programa com o nome nomedoprograma. Basta, agora, executá-lo — e você escreveu seu primeiro programa em C! Parabéns!



Link

Além de IDEs, existem outras ferramentas para otimizar o desenvolvimento de software. Para maiores detalhes, acesse o link a seguir ou o código QR.

<https://goo.gl/79C11q>



Leitura Recomendada

PINHEIRO, F. A. C. *Elementos de programação em C*. Porto Alegre: Bookman, 2012.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.

Conteúdo:



SOLUÇÕES
EDUCACIONAIS
INTEGRADAS

Dica do Professor

O que é uma linguagem de programação? De que é formada uma linguagem de programação e quais os conceitos básicos da linguagem C?

Acompanhe o vídeo da Dica do Professor para receber orientações sobre esses conceitos e se familiarizar com a linguagem C.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Exercícios

- 1) Sobre a linguagem C, é correto afirmar que:
- A) sua portabilidade se dá pelo fato de que pode-se rodar um programa executável em qualquer sistema operacional, sem precisar recompilar.
 - B) os programas escritos em C têm alto grau de segurança.
 - C) essa linguagem tem características que permitem o uso do paradigma de orientação a objetos.
 - D) não tem recursos de baixo nível, ou seja, de acesso direto ao *hardware*.
 - E) é uma das linguagens indicadas para se começar a estudar programação, pois é considerada simples e bem-estruturada.
- 2) O seguinte programa está escrito em linguagem C. Qual a função da linha 1?

```
1  #include <stdio.h>
2  int main()
3  {
4      printf("Bem-vindo ao seu primeiro Jogo em C!\n");
5      return 0;
6  }
```

- A) Acrescentar a biblioteca padrão da linguagem C para que se possa fazer entrada e saída de dados.

- B) Incluir um tipo de arquivo de dados no programa.
- C) Essa instrução pode ser retirada de acordo com o que está se desejando escrever no programa.
- D) Não pode ser usada se serão feitas entrada e saída de dados.
- E) Permite que mais de um usuário utilize o código.

3) O que é uma linguagem de programação?

- A) É a linguagem utilizada pelos programadores para realizar documentação sobre os programas que eles escrevem.
- B) É a linguagem falada pelos humanos, mas, em vez de uma conversa entre humanos, trata-se de uma conversa entre humanos e um computador.
- C) É a linguagem utilizada para escrever programas de computadores.
- D) É a linguagem na qual os algoritmos são escritos.
- E) É a linguagem composta por 0 e 1 e impossível de ser utilizada por um ser humano.

4) Utilizando o editor e o compilador, quais os passos para que se tenha um programa em C pronto para execução?

- A) Escrever o programa em um arquivo, salvar o arquivo com extensão C e executá-lo na linha de comando.
- B) Escrever o programa em um arquivo com extensão txt, com a seguinte linha de comando: gcc -o nomedoprograma.txt nomedoexecutavel.
- C) Escrever o programa em um arquivo, salvar o arquivo com extensão c e compilar com a seguinte linha de comando: gcc nomedoprograma.c -o nomedoexecutavel.
- D) Não é possível desenvolver um programa em C somente com editor de texto e compilador.
- E) Escrever o programa em um arquivo com qualquer extensão e compilar com a seguinte linha de comando: gcc -o nomedoprograma nomedoexecutavel.

5) A linguagem C é uma linguagem:

- A)** de médio nível e compilada.
- B)** de alto nível e interpretada.
- C)** de baixo nível e compilada.
- D)** de baixo nível e interpretada.
- E)** Nenhuma das alternativas anteriores.

Na prática

Taís está em seu primeiro dia de trabalho no setor de desenvolvimento de sistemas em uma empresa de alta tecnologia. Seu primeiro dia se dá junto com o primeiro dia de estágio de Felipe, o estagiário da equipe de apoio.

FELIPE

Olá, Taís. Sou Felipe, o estagiário da equipe de apoio. Pode me encaminhar uma lista dos *softwares* necessários para iniciar seu trabalho?

TAÍS

Claro, Felipe. Além dos *softwares* de escritório (editor de texto, planilha...) preciso de uma IDE.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

LOGO DEPOIS, TAÍS RECEBE A SEGUINTE SOLICITAÇÃO DA EMPRESA POR E-MAIL:



Olá Taís, bem-vinda a nossa equipe!

Gostaríamos de solicitar que, em vez de ter um novo software, você utilize somente o compilador e um editor de texto da linguagem de programação, da mesma forma que utilizará no desenvolvimento do sistema. Em caso negativo, nos encaminhe sua justificativa.

Atenciosamente,
Equipe de Apoio

TAÍS RESPONDE O E-MAIL:



Olá, Equipe de Apoio.

Solicito um ambiente de desenvolvimento integrado (IDE) para ter as seguintes ferramentas em um ambiente único: editor de texto, compilador/interpretador da linguagem e ferramenta para debugar.

A fase de debugar é muito importante quando se fala em produtividade em ambientes de produção de sistemas computacionais.

Além disso, há algumas vantagens no uso de uma IDE, como: aumento de produtividade, diminuição de gastos e aumento de desempenho.

Atenciosamente,
Taís

Saiba mais

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

Linguagens mais Populares de 2017

Com tantas linguagens de programação no mercado, é interessante estar atento às linguagens de programação mais populares do ano para organizar os estudos. Neste video, você verá as linguagens de programação mais populares do ano de 2017.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

O que significa "COMPILAR" e quando você precisa fazer isso no Linux

O vídeo mostra como se dá o processo de compilação de um programa no sistema operacional Linux.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Dia do Programador: alguns GRANDES Programadores da História

Quem foi a primeira pessoa a programar um computador? Quais foram as pessoas que começaram nesse universo incrível de programação? Neste vídeo, você vai conhecer alguns grandes programadores da história.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

O que eu preciso saber para virar engenheiro de software na Google?

A Google é uma das principais empresas de tecnologia e um dos grandes sonhos de emprego de muitos dos estudantes de computação. Quer saber o que você precisa fazer para se tornar um engenheiro de software da Google? Acesse o artigo e fique por dentro.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.