Começando a Programar em C Para Leigos

Folha de Cola

A melhor maneira de aprender a programar é começar com uma linguagem fundamental como C. Quase todas as outras linguagens populares de hoje pegam algo emprestado de C. Caso você seja curioso sobre programação, precisa passar em um curso da faculdade ou quer começar seu próprio negócio de aplicativos, aprender C é o jeito certo de começar.

Entendendo o Esqueleto de Linguagem C

A maioria das codificações começam com uma estrutura de linguagem C. Este *esqueleto* inclui os ossos básicos sobre os quais a maioria dos programas são escritos. Use este simples esqueleto para começar:

```
#include <stdio.h>
int main()
{
return(0);
}
```

Tradicionalmente, o programa começa com diretivas de pré-processamento mais protótipos. As declarações `#include` trazem arquivos de cabeçalho, como `stdio.h`, o arquivo cabeçalho padrão de entrada/saída.

A função primária em todo o código C é `main()`, que é a primeira função que executa quando o programa começa. A função `main()` é uma função `int`, então deve retornar um valor inteiro. Todas as declarações de funções estão inclusas entre chaves.

Palavras-chave de Linguagem C

As palavras-chave de linguagem C representam a base da linguagem. Com a revisão C11 para a linguagem, várias palavras-chave novas foram adicionadas. Elas são exibidas iniciando com o sublinhado (*underscore*) na tabela seguinte:

_Alignas	break	float	signed
_Alignof	case	for	sizeof
_Atomic	char	goto	static
_Bool	const	if	struct
_Complex	continue	inline	switch
_Generic	default	int	typedef
_Imaginary	do	long	union
_Noreturn	double	register	unsigned
_Static_assert	else	restrict	void
_Thread_local	enum	return	volatile
auto	extern	short	while

Lembre-se dos seguintes pontos quando começar a programar em C:

- Não nomeie nenhuma função ou variável da mesma maneira que uma palavra-chave.
- Você usa apenas algumas palavras-chave de linguagem C no seu código. Algumas delas você provavelmente nunca usará.
- A maioria do trabalho no seu código é feita por funções, não por palavras-chave.

Lembre-se dessas observações sobre tipos de variáveis:

- Garanta que você escolha o tipo de variável adequado para os valores que você precisa armazenar.
- O tipo `_Bool` armazena apenas dois valores, 0 e 1, que podem representar VERDADEIRO ou FALSO ou Ligado e Desligado em qualquer condição binária.
- A variável tipo `char` armazena valores de caracteres, embora também possa ser usada para armazenar inteiros.
- Inteiros, ou números inteiros, são armazenados nas variáveis tipo `int`.
- Qualquer tipo de valor, do maior para o menor, e qualquer valor fracionário são armazenados nos tipos `float` e `double`.
- Lembre-se de usar valores `int` para funções que geram inteiros, como `getchar()`. É fácil supor que a função retorna um valor `char` por causa do nome da função.
- C não possui um tipo de variável string. Em vez disso, um array de variáveis `char` é usado.
- Outros tipos de variáveis incluem estruturas e ponteiros.

Sequências de Escape Comuns em C

Quando você não pode digitar caracteres em sua string, use as sequências de escape para inserir caracteres não-imprimíveis em suas strings de texto, variáveis char e arrays. Aqui estão algumas sequências de escape em C:

Caracteres	O que Representa ou Exibe
\a	Alarme (“beep!”)
\b	Barra de espaço, não destrutivo
\f	Feed de formulário ou limpa a tela
\n	Newline (nova linha)
\r	Código de fim de linha
\t	Tab
\v	Tab vertical
\\	Caractere de contrabarra
\?	Ponto de interrogação
\’	Aspas simples
\”	Aspas duplas
\xnn	Código de caractere hexadecimal nn
\onn	Código de caractere octal nn
\nn	Código de caractere octal nn

Começando a Programar em C

PARA

LEIGOS®

por Dan Gookin

ALTA BOOKS
EDITORA
Rio de Janeiro, 2016

Sobre o Autor

Dan Gookin escreve sobre tecnologia há mais de 25 anos. Ele combina seu amor por escrever com a fascinação por dispositivos para criar livros que são informativos, interessantes e divertidos. Já tendo escrito mais de 130 títulos, com 12 milhões de cópias impressas traduzidas em mais de 30 línguas, Dan pode provar que seu método de criar livros de computação funciona.

Talvez, seu título mais famoso seja o original *DOS For Dummies*, publicado em 1991. Ele se tornou o livro de computação de venda mais rápida do mundo, vendendo mais cópias por semana de uma só vez que o best-seller número um do *New York Times* (embora, como livro de referência, ele não possa ser listado na Lista de Best Sellers do jornal). Esse livro gerou uma linha inteira de livros *For Dummies*, que se mantém como fenômeno de publicação até os dias de hoje.

Os títulos mais populares de Dan incluem *PCs Para Leigos, Notebooks e Laptops Para Leigos, Word For Dummies* e *Android Phones For Dummies*.

Dan possui graduação em Comunicação/Artes Visuais pela Universidade da Califórnia, San Diego. Mora no noroeste Pacífico, onde gosta de passar o tempo com seus filhos, jogando videogames dentro de casa, enquanto apreciam as madeiras suaves de Idaho.

Agradecimentos

Gostaria de agradecer a Jon Rossen por seu feedback e revisão deste livro. Jon tem lido meus livros de C há muito tempo. De vez em quando, ele me mandava um e-mail com uma pergunta ou sugestão. Eu aproveitei nossa comunicação e o jeito que ele olha as coisas e é por isso que estou muito agradecido por sua revisão. O resultado final foi muito melhor do que seria sem ele. Jon, obrigado por suas contribuições.

Sumário Resumido

Sumário

Introdução

Diga "Olá" para *Começando a Programar em C Para Leigos*, um livro que transforma você, um ser humano afetuoso e bem-intencionado, em um elemento admirado da subcultura nerd underground: um programador.

Ah, sim, isso é uma coisa boa. Ao aprender a codificar em C, você se torna o mestre final de muitos dispositivos eletrônicos. Passa a poder criar seus próprios programas, ditando, para computadores, tablets e telefones celulares, seus caprichos e desejos. Os eletrônicos obedecem prontamente! Com as informações oferecidas neste livro, você pode esquecer aquela aula de programação, impressionar seus amigos, ser admirado por Hollywood ou, até mesmo, começar sua própria empresa de software. Sim, aprender a programar é um investimento valioso do seu tempo.

Este livro torna o aprendizado da programação compreensível e agradável. Você não precisa de nenhuma experiência em programação — você nem precisa comprar um software novo! Você só precisa querer programar em C e ter habilidade para se divertir enquanto faz isso.

A Linguagem C é Relevante?

A cada poucos anos, o argumento de que aprender C não leva a nada, vem à tona. Existem linguagens de programação novas e melhores e é preferível aprendê-las a perder tempo aprendendo C. Bobagem.

De certa maneira, C é o Latim das linguagens de programação. Praticamente todas as linguagens de programação recentes usam a sintaxe C. Palavras-chave de C e até certas funções encontram seus lugares em outras linguagens populares, de C++, passando pelo Java, para Python e para qualquer outra linguagem que estiver na moda.

Minha opinião é que, uma vez que se aprenda a programar em C, todas as outras linguagens de programação serão mais fáceis. Na verdade, muitos dos livros que ensinam essas outras linguagens frequentemente presumem que o leitor sabe um pouco de C antes de começar. Isto pode ser frustrante para qualquer principiante — mas não quando você já sabe C.

Apesar do que os especialistas e sábios dizem, C ainda é relevante. A programação para microcontroladores, sistemas de operação e importantes pacotes de software ainda é feita utilizando-se a boa e velha C. Não é perda de tempo.

A Abordagem de Começando a Programar em C Para Leigos

Como programador, trabalhei com muitos e muitos livros de programação. Sei do que não gosto de ver, mas lamentavelmente vejo com muita frequência autores escreverem códigos de várias páginas ou simplesmente se gabarem pelo que sabem, impressionando seus companheiros nerds e não ensinando nada de fato. Existe muito desse tipo de treinamento e é provavelmente por causa disso que você escolheu este livro. Minha abordagem aqui é simples: programas curtos. Demonstrações diretas. Muitos exemplos. Exercícios de sobra.

O melhor jeito de aprender alguma coisa é fazendo. Cada conceito apresentado neste livro é ligado a um exemplo de código. As listagens são curtas o suficiente para que você possa incluí-las rapidamente — e recomendo que faça isso. Depois, você pode montar e executar o código para ver como as coisas funcionam. Esse feedback imediato não é só gratificante, mas também uma ferramenta de aprendizado maravilhosa.

Exemplos de programas são seguidos por exercícios, que você pode tentar resolver sozinho, testando suas habilidades e expandindo seu conhecimento. Sugestões de respostas para os exercícios podem ser encontradas no site relacionado a este livro:

```
http://www.c-for-dummies.com/begc4d/exercises
```

Como Este Livro Funciona

Este livro ensina programação em linguagem C. Começa assumindo que você sabe de muito pouco a nada sobre programação e termina cobrindo algumas das operações mais avançadas em C.

Para programar em C você precisa de um computador. Não há indicação sobre o computador "ideal" que você deve escolher: pode ser um PC com Windows, um Macintosh ou um sistema Linux. Importante para todos os sistemas, e para programar com este livro, é a configuração e utilização do ambiente de desenvolvimento integrado (ou IDE) Code::Blocks. Os passos para isso serão oferecidos no Capítulo 1.

Este livro também não perde tempo, fazendo você programar já no Capítulo 1. Nada é introduzido sem uma explicação completa antes, embora, devido à natureza da programação, eu tenha feito algumas exceções, que estão cuidadosamente anotadas no texto. Ademais, a melhor maneira de ler este livro é da primeira página ao seu final.

Palavras-chave e funções da linguagem C são mostradas em fonte monoespaçada, como `printf()` e `break`. Algumas palavras-chave, como `for` e `if`, podem fazer as frases serem lidas erroneamente, de uma forma esquisita, e é por isso que são exibidas em fonte monoespaçada.

Nomes de arquivos são mostrados em fonte monoespaçada, como `program.exe`.

Se você precisar digitar alguma coisa, esse texto será mostrado em negrito. Por exemplo, "Digite o comando **blorfus**" significa que você deve digitar `blorfus` no teclado. Você é orientado quanto ao momento de pressionar a tecla Enter, se precisar.

Quando trabalharmos com passos numerados, o texto para digitar aparecerá em um formato regular (roman):

3. **Digite** exit **e pressione a tecla Enter.**

Você digita a palavra *exit* e então pressiona a tecla Enter.

Exemplos de programas são mostrados em fragmentos na página, parecidos com este:

```
if( i == 1)
    printf("eye won");
```

Você não precisa digitar um exemplo a não ser que seja orientado a fazê-lo.

Listagens completas de programas são mostradas e numeradas em cada capítulo; por exemplo:

Listagem 1-1: O Esqueleto do Code::Blocks

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    printf("Hello world!\n");
    return(0);
}
```

Por causa das margens deste livro, o texto de uma listagem pode ser quebrado, passando de uma linha para a outra. Você não precisa dividir o seu código de uma maneira similar e será lembrado disso sempre que esse tipo de coisa ocorrer.

As listagens neste livro não contêm linhas numeradas, mas no editor no Code::Blocks sim (como em vários editores de texto populares). Este livro referencia as listagens de exemplos de código usando o número das linhas. Utilize os números das linhas em seu editor para referenciar o código.

Exercícios são numerados por capítulo e então sequencialmente. Logo, o terceiro exercício do Capítulo 13 é o Exercício 13-3. Você é orientado no texto a trabalhar em um exercício. Aqui está um exemplo:

Exercício 1-1: Digite o código-fonte da Listagem 1-1 no editor Code::Blocks. Salve com o nome de arquivo `ex0101`. Monte e execute.

Respostas para todos os exercícios podem ser encontradas nesse site (conteúdo em inglês):

```
http://www.c-for-dummies.com/begc4d/exercises
```

Vá nesta página, se quiser copiar e colar o código-fonte também; não ofereço os arquivos de código-fonte porque você aprende melhor quando digita os exercícios. Vou conceder a você um "copiar e colar", mas, sinceramente, o código-fonte é tão curto que você pode digitá-lo rapidamente.

Ícones Utilizados Neste Livro

Este ícone assinala informações que valem a pena lembrar. Embora recomende que você grave o máximo que puder, estes ícones marcam as coisas que simplesmente não se pode esquecer.

Uma dica é uma sugestão, um truque especial ou algo super sofisticado para ajudá-lo.

Este ícone marca algo que você deve evitar. É um conselho que também poderia ser assinalado com um ícone de Dica ou Cuidado, mas tem consequências terríveis se ignorado.

Encare o fato: toda a programação é técnica. Reservo o uso deste ícone para pedacinhos extratécnicos, exceções e anedotas. Chame-os de "coisas nerds".

Pensamentos Finais

Gosto de programar. É um hobby, que acho incrivelmente relaxante, frustrante e recompensador. Presumo que não mais do que apenas algumas pessoas compartilhem destes sentimentos, mas você também pode ser um estudante com dificuldades ou alguém que almeja uma carreira. De qualquer maneira, *goste* de programar. Se você pode imaginar o programa que quer escrever em uma tela, você pode fazer acontecer. Pode não acontecer tão rápido quanto gostaria, mas pode acontecer.

Por favor, faça os exercícios deste livro. Tente alguns próprios, variações de um tema. Continue trabalhando com problemas até resolvê-los. O incrível da programação é que não existe um jeito único absolutamente correto de fazer alguma coisa. Sempre que tentar, aprenderá.

Se possível, encontre um amigo programador que possa ajudá-lo. Não o faça fazer o trabalho por você ou explicar como as coisas funcionam, mas conte com ele como um recurso. Programar pode ser uma coisa individual, mas ocasionalmente é bom solidarizar-se com outros que também programam em C — ou qualquer outra linguagem.

Este livro possui alguns sites que se relacionam a ele.

Para analisar as respostas dos exercícios, encontrar informações suplementares e ver o que há de novo, visite os seguintes sites com conteúdo em inglês:

```
www.c-for-dummies.com
```

Para arquivos de códigos do livro, visite o site da editora e procure pelo título do livro:

```
www.altabooks.com.br
```

Para algumas dicas úteis e artigos de Programação em C, acesse e procure pelo título do livro (conteúdo em inglês):

```
www.dummies.com
```

Aproveite sua programação em C!

Parte I

Nesta parte...

- Comece com a IDE Code::Blocks
- Trabalhe no seu primeiro programa
- Aprenda como a programação funciona
- Descubra as partes da linguagem C
- Use o Code::Blocks para escrever o esqueleto básico de C

Capítulo 1

Um Início Rápido Para os Impacientes

Neste Capítulo

- Obtendo o Code::Blocks
- Configurando seu primeiro projeto
- Digitando em código
- Montando e executando
- Saindo do Code::Blocks

Você deve estar ansioso para começar a programar em C. Eu não farei com que perca tempo.

O Que Você Precisa para Programar

Para ter total controle sobre um computador, tablet, telefone celular, console de vídeogame etc, você precisa de algumas ferramentas de software. A boa notícia é que, a esta altura do século 21, todas essas ferramentas são gratuitas e fáceis de serem obtidas através da internet. Você só precisa saber o que é necessário e onde conseguir.

Obtendo ferramentas de programação

As duas coisas mais importantes que você precisa para começar sua aventura na programação são:

- um computador
- acesso à internet

O computador é sua ferramenta primária para escrever e compilar código. Sim, até mesmo se estiver escrevendo um jogo para o Xbox, você precisará de um computador para poder codificar. O computador pode ser um PC ou um Macintosh. O PC pode executar Windows ou Linux.

O acesso à internet é necessário para obter o restante das suas ferramentas de programação. Você precisará de um editor de texto para escrever o código de programação e um compilador para traduzir o código em um programa. O compilador geralmente vem com outras ferramentas necessárias, como um vinculador e um depurador. Todas essas ferramentas são encontradas sem custo na internet.

Não pire! Os termos *compilador*, *vinculador* e *depurador* estão todos definidos no Capítulo 2.

Adquirindo um Ambiente de Desenvolvimento Integrado (IDE)

É ótimo que você comece sua jornada na programação, procurando por um editor de texto, um compilador e outras ferramentas. Utilizar programas diferentes na linha de comando de uma janela de terminal foi como aprendi a programar, lá na idade das trevas. É um processo que ainda pode ser feito, mas é estranho.

O jeito mais tranquilo e profissional de codificar atualmente é através de um Ambiente de Desenvolvimento Integrado — popularmente chamado de *IDE*. Ele combina todas as ferramentas necessárias para programar em uma unidade compacta, aterrorizante e intimidadora.

Você utiliza o IDE para escrever código, montar e depurar programas, além de encontrar todos os tipos de magia. Apesar de ser nostálgico utilizar um editor de texto ou um compilador separados, todos os profissionais utilizam um IDE. É isso que recomendo neste livro também.

Obter um IDE também resolve o grande número de problemas de configurar um compilador, o editor de texto e fazer todos esses elementos separados trabalharem juntos. Ter um IDE é a melhor maneira de começar a programar — que, acho, é algo que você está realmente ansioso para fazer.

Eis o IDE Code::Blocks

Você vai ver que a internet está cheia de vários IDEs e são todos muito bons. Para ser consistente, este livro usa o IDE Code::Blocks, que funciona no Windows, no Mac OS C e no Linux. Ele vem com tudo o que você precisa.

Se você já tem um IDE, ótimo! Estou certo de que ele faz coisas similares ao Code::Blocks, embora as ilustrações e exemplos deste livro, especialmente nos primeiros capítulos, sejam específicos do Code::Blocks.

Instalando o Code::Blocks

Obtenha o Code::Blocks na internet neste site:

```
www.codeblocks.org
```

Este site certamente será modificado com o tempo. Então, os seguintes passos para instalar o IDE podem mudar sutilmente:

1. **Use o navegador do seu computador para visitar o site do Code::Blocks.**

2. **Entre na área de Downloads.**

   Baixe a versão binária ou a versão executável do Code::Blocks. Ela deve ser específica para o sistema operacional do seu computador. Mais para frente, encontre a versão que inclui um compilador C, como o compilador comum *MinGW*.

3. **Clique no link para exibir a instalação do binário ou do executável do Code::Blocks.**

   No momento deste livro ir para a gráfica, o link estava na frase *Download the Binary Release*.

4. **Escolha o sistema operacional do seu computador ou role a tela até a parte que lista as opções para esse sistema operacional.**

   Você pode encontrar seções (ou páginas) para Windows, Linux e Mac OS X.

5. **Clique no link que baixa o compilador e o IDE para o sistema operacional do seu computador.**

   A versão para Windows do IDE e do compilador está nomeada desta forma:

```
codeblocks-xx.yymingw-setup.exe
```

   O *xx* e o *yy* representam o maior e menor números de versões do Code::Blocks.

   No Linux, você pode escolher a versão 32-bit ou a 64-bit, dependendo da sua distribuição, ou *distro*, e o formato de arquivo que você quer. Recomendo uma versão estável.

   Usuários de Mac OS X podem escolher entre baixar um arquivo `.dmg`, ou uma imagem de disco, ou o arquivo zip tradicional.

6. **Extraia o programa de instalação do Code::Blocks do arquivo.**

   Apesar do título *Para Leigos* deste livro, parto do princípio que você sabe trabalhar bem com arquivos `.zip`, `.gz`, `.dmg`, e outros formatos para qualquer que seja o sistema operacional que utilize.

7. **Execute o programa de instalação.**

   Observe com atenção as orientações na tela. Realize uma instalação padrão; você não precisa customizar nada a essa altura.

   No Windows, garanta que você esteja instalando o *MinGW compiler suite*. Se você não vê essa opção presente na janela *Choose Components*, então baixou a versão errada do Code::Blocks. Volte ao Passo 5.

8. **Termine a instalação executando o Code::Blocks.**

   No meu computador, surgiu um lembrete perguntando se eu queria executar o Code::Blocks. Cliquei no botão *Yes*. Se você não visualizar este lembrete, utilize o sistema operacional do computador para inicializar o Code::Blocks do mesmo jeito que faria para inicializar qualquer programa.

9. **Feche a janela de instalação.**

   Mesmo que veja o Code::Blocks espalhado por toda a tela, você ainda precisa finalizar, fechando a janela de instalação.

A próxima seção oferece uma visão geral da interface do programa Code::Blocks.

Um tour pela área de trabalho do Code::Blocks

Se o Code::Blocks não inicializou, vá em frente e o inicialize, como você faria com qualquer outro programa: localize o ícone no menu Iniciar ou encontre o ícone de atalho para o Code::Blocks na Área de Trabalho, que é o jeito mais fácil de inicializar o IDE no Windows 8.

A Figura 1-1 ilustra a *área de trabalho* do Code::Blocks, que é o nome oficial do mosaico enorme de janelas que você vê na tela. Os detalhes da Figura 1-1 são bem pequenos, mas o que você precisa encontrar são as áreas principais, que estão na figura e são chamadas:

Toolbars (Barra de ferramentas): Essas tiras bagunçadas, adornadas com vários botões de comando, agarram-se ao topo da janela do Code::Blocks. Existem oito barras de ferramentas, que você pode rearranjar, exibir ou esconder. Não mexa com elas até se sentir confortável com a interface.

Management (Gerenciamento): A janela do lado esquerdo da área de trabalho possui quatro abas, embora você não veja todas ao mesmo tempo. A janela fornece uma visão geral conveniente dos seus esforços de programação.

Status Bar (Barra de status)**:** Na parte inferior da tela, você vê informações sobre o projeto e editor, e sobre outras atividades que possam ocorrer no Code::Blocks.

Editor: A janela grande na área central direita da tela é onde você digita o código.

Logs: A parte inferior da tela possui uma janela com muitas, muitas abas. Cada aba exibe informações sobre seus projetos de programação. A aba que você utilizará mais frequentemente é chamada de *Build Log*.

O menu Visualizar controla a visibilidade de cada item exibido na janela. Escolha o comando adequado, como o *Manager*, do menu *View* para exibir

ou esconder aquele item. Controle as barras de ferramentas, utilizando o submenu *View ⇨Toolbars*.

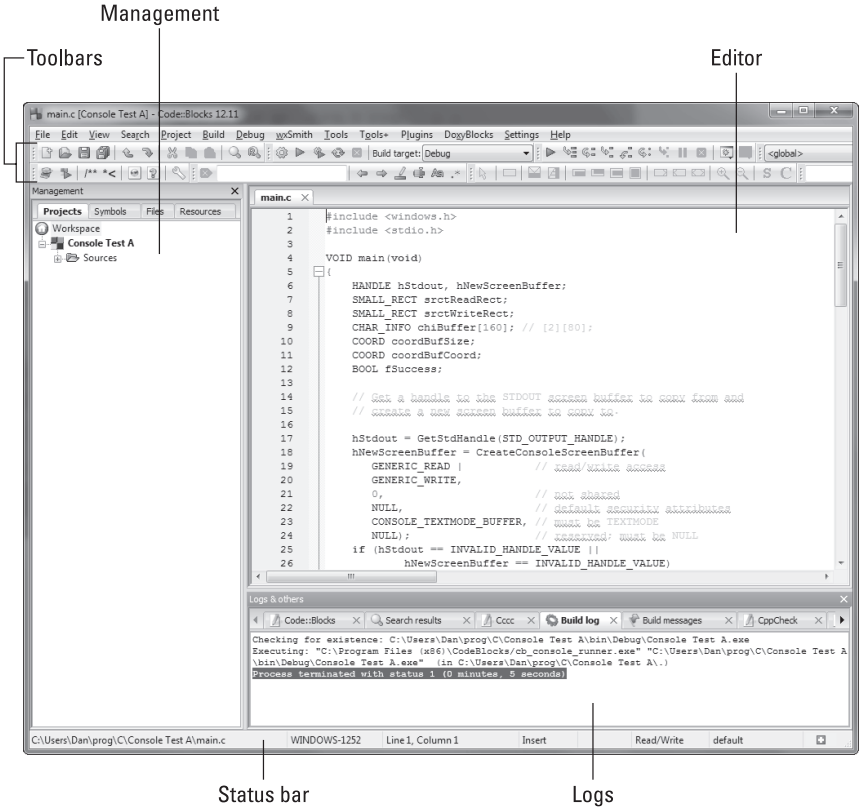

Figura 1-1: A área de trabalho do Code::Blocks.

A coisa mais importante a ser lembrada sobre a interface do Code::Blocks é não ficar confuso com ela. Um IDE como o Code::Blocks pode ser altamente intimidante, mesmo quando você se considera um veterano em programação. Não se preocupe: você logo se sentirá em casa.

- Maximize a janela do programa Code::Blocks para que preencha a tela. Você precisará de toda essa área.
- Cada uma das várias áreas na tela — Management, Editor, Logs — pode ser redimensionada: posicione o ponteiro do mouse entre duas áreas. Quando o ponteiro virar uma "coisinha" de flechas duplas, você pode arrastar o mouse para mudar o tamanho da área.
- As áreas do Editor e de Logs possuem interfaces com abas. Cada janela exibe diversas "folhas" de informações. Troque entre as folhas, escolhendo uma aba diferente.

Seu Primeiro Projeto

Em programação tradicional para computador, utilizava-se um editor de texto, um compilador e um vinculador separados. A ação acontecia no *prompt de comando*, onde você digitava os comandos para editar, compilar e vincular. Era um processo linear e funcionava muito bem para pequenos projetos. Com o advento dos sistemas operacionais modernos e da programação para dispositivos móveis e consoles de vídeogame, esse método linear tornou-se altamente ineficiente.

O IDE moderno ainda tem elementos de um editor, um compilador, um vinculador, um depurador e de outras ferramentas de programação. Exibe características necessárias para criar programas gráficos e projetos complexos. Por essa razão, o IDE é orientado a trabalhar com projetos e não apenas com programas individuais.

Esta seção explica o processo de criação de um projeto, utilizando-se o IDE Code::Blocks.

Criando um novo projeto

Os exemplos apresentados neste livro são todos aplicações de *console*, o que significa que eles executam em modo Texto em uma janela de terminal. Sinto que esta é a melhor maneira de ensinar conceitos básicos de programação sem sobrecarregar com um programa grande, complexo e graficamente complicado. Então, apesar de um IDE ser capaz de mais, você o utiliza, neste livro, para criar programas simples e baseados em console. É assim que funciona:

1. **Inicialize o Code::Blocks.**

   Você vê a Tela *Start Here*, que exibe a logo do Code::Blocks e alguns links. Caso não veja a Tela Start Here, selecione *File* ⇨ *Close Workspace*.

2. **Clique no link *Create a New Project*.**

   A caixa de diálogo *New from Template* vai aparecer, como mostrado na Figura 1-2.

3. **Escolha *Console Application* e então clique no botão *Go*.**

   O Assistente de Console Application vai aparecer.

   

   Você pode inserir uma marca de verificação ao lado do item *Skip This Page Next Time* para pular a primeira tela do assistente.

4. **Clique no botão *Next*.**

5. **Escolha C como a linguagem que você quer utilizar e então clique no botão Next.**

   C é bem diferente de C++ — você pode fazer coisas em uma linguagem que não são permitidas na outra.