ANTÓNIO TRIGO • JORGE HENRIQUES

Aprenda a Programar com C#

- C# 7.2
- Microsoft Visual Studio 2017
- Aprendizagem passo a passo
- +100 exemplos
- +200 exercícios resolvidos
- Todo o código disponível *online*

EDIÇÕES SÍLABO

Aprenda a programar com C#

ANTÓNIO TRIGO
JORGE HENRIQUES

EDIÇÕES SÍLABO

Visite a Sílabo na rede

www.silabo.pt

FICHA TÉCNICA:

Título: Aprenda a programar com C#
Autores: António Trigo, Jorge Henriques
© Edições Sílabo, Lda.
Capa: Pedro Mota

1ª Edição – Lisboa, janeiro de 2018
Impressão e acabamentos: Cafilesa – Soluções Gráficas, Lda.
Depósito Legal: 436226/18
ISBN: 978-972-618-934-3

EDIÇÕES SÍLABO, Lda.
Publicamos conhecimento

Editor: Manuel Robalo

R. Cidade de Manchester, 2
1170-100 Lisboa
Tel.: 218130345
e-mail: silabo@silabo.pt
www.silabo.pt

Índice

Prefácio

Este livro foi pensado para quem nunca programou, para quem tem curiosidade em saber o que é a programação e deseja dar os primeiros passos. É um livro de iniciação à algoritmia e programação não sendo necessário o conhecimento prévio de uma linguagem de programação, somente estar à vontade com um computador.

No primeiro capítulo é feita uma breve introdução à algoritmia, programação, linguagem C# e plataforma .NET, para a qual irão ser desenvolvidos os programas apresentados neste livros.

No segundo capítulo ensina-se a instalar o ambiente de desenvolvimento e apresentam-se os primeiros programas, simples, mas com exemplos das operações base de qualquer programa que são, a entrada e saída de dados e o processamento dos mesmos, recorrendo a variáveis, operações de atribuição e operações aritméticas. Espera-se neste segundo capítulo, que o estudante seja capaz, de forma autónoma, de implementar os programas apresentados e compreender como se executa e depura um programa a partir do ambiente de desenvolvimento.

Nos capítulos seguintes, do terceiro ao nono capítulo são apresentados os conceitos estruturantes da programação estruturada, que inclui os tipos de dados, variáveis, constantes, operadores, entrada e saída de dados, decisões, repetições, métodos, vetores, matrizes e texto. No capítulo décimo é apresentado o tratamento de erros em C#. O capítulo seguinte apresenta a implementação de um jogo com base nos conceitos anteriores. Por fim, o último capítulo, de introdução ao paradigma de programação orientada aos objetos, apresenta ao estudante conceitos base deste paradigma que lhe permitem prosseguir o estudo da linguagem C#.

Ao longo do livro serão apresentados exemplos relativos à matéria em questão. Sempre que os exemplos possuírem algoritmos que devam ser me-

morizados, como por exemplo, a contagem ou a soma de um conjunto de números, os mesmos serão classificados de memorizar. A necessidade de fazer esta diferença prende-se com o facto de que estes exemplos a memorizar poderem ser utilizados na resolução de outros exercícios. A título de exemplo, o algoritmo da soma pode ser utilizado tanto para somar um conjunto de números introduzidos pelo utilizador, como para somar todos os números existentes num vetor. Também neste sentido da memorização existem ao longo do livro chamadas de atenção, marcadas com o símbolo de perigo (⚠), que servem para realçar coisas importantes, como truques ou confusões frequentes a evitar, como a utilização dos operadores "=" e "==".

Na maior parte dos exemplos a visualização dos resultados/saídas, em inglês *outputs*, do código terá de ser feita pelo estudante, devido a restrições de espaço no livro. Só em alguns casos particulares como nos capítulos "O meu primeiro programa" ou "Leitura e escrita de dados" é que serão colocados os resultados/saídas dos programas para visualização das questões de escrita formatada.

Dado que o objetivo deste livro é iniciar o estudante, que nada entende de programação, solicita-se que, sempre que encontre algo menos claro, ou que tenha mais dificuldade em seguir, envie um e-mail aos autores: antonio.trigo@gmail.com; jmvhenriques@gmail.com. Sugestões de melhoria também são bem vindas!

Os exemplos apresentados neste livro e cujo código está disponível *online* na plataforma *github* ou na página online do livro em `http://www.silabo.pt` podem ser utilizados, agradecendo os autores, que citem esta obra quando o fizerem.

Espera-se que com este livro entre facilmente no mundo da programação.

Lista de exemplos

Capítulo 8

Capítulo 9

Capítulo 10

Lista de exemplos a memorizar

Capítulo 1

Introdução

Neste capítulo de introdução são apresentados alguns conceitos base relativos à programação, à linguagem C# e à plataforma *.NET*. Caso queira passar já à ação pode saltar diretamente para o capítulo 2 e à medida que for necessitando de alguns dos conceitos aqui apresentados voltar a este capítulo.

1. Programar, o que é?

Um livro dedicado a aprender a programar deve começar por explicar o que é programar. Pois bem, programar é dizer ao computador o que é que ele deve fazer, ou seja, é dar ao computador instruções, de uma forma lógica e sistemática. O que resulta de programar ou da atividade de programação é o programa, que é um conjunto de instruções que descrevem uma tarefa a ser realizada pelo computador.

Ao começar a escrever programas vai ganhar uma nova competência que é a de colocar o computador a fazer coisas para si escritas por si e não somente utilizá-lo numa ótica do utilizador.

A atividade de programação é um processo constituído tipicamente pelas seguintes fases:

- Análise do problema, em que se tenta da forma mais clara possível definir o problema a resolver.
- Conceção, em que se procura a estratégia mais adequada para resolver o problema, com a criação do respetivo **algoritmo**.
- Codificação, que consiste na escrita do algoritmo numa determinada linguagem, da qual resulta o **programa**.
- Teste, em que se testa se o programa escrito resolve o problema. Nesta fase é comum verificar se o programa produz as saídas expectáveis para determinadas entradas.
- Manutenção, que diz respeito a alterações ou aperfeiçoamentos que se desejem introduzir no programa.

1.1. Algoritmo

Um algoritmo consiste numa sequência finita de ações que conduzem à resolução de um problema. Tipicamente um algoritmo transforma um conjunto de entradas em saídas, ou seja, recebe um conjunto de dados, aplica-lhe operações, como cálculos e depois apresenta os resultados (saídas).

De seguida apresentam-se dois algoritmos, um para a mudança dos pneus de um carro e outro para o cálculo da área de circulo.

Algoritmo para troca de um pneu:

1. Abrir a mala do carro
2. Retirar o pneu sobresselente e as ferramentas para levantar o carro

3. Colocar o macaco na posição adequada
4. Desapertar um pouco as porcas
5. Levantar o carro
6. Desapertar totalmente as porcas
7. Trocar o pneu
8. Apertar um pouco as porcas
9. Baixar o carro
10. Apertar totalmente as porcas
11. Arrumar as ferramentas e o pneu sobresselente na mala
12. Fechar a mala

Algoritmo para o cálculo da área do círculo:

1. Pedir o raio do círculo
2. Calcular a área do círculo
3. Mostrar a área do círculo

Caso deseje ver o programa que implementa este segundo algoritmo pode consultar o terceiro capítulo deste livro (exemplo 3.3).

Os algoritmos podem ser descritos em linguagem natural, pseudocódigo ou ainda através de fluxogramas, como os apresentados nos capítulos das decisões e repetições.

2. Linguagem C#

Tal como para comunicarmos uns com os outros utilizamos a linguagem, traduzida no nosso caso na utilização da língua portuguesa, também para dar instruções ao computador é preciso utilizar uma linguagem, neste caso a linguagem C#.

Esta linguagem foi criada pela Microsoft para ser a linguagem de desenvolvimento de aplicações para o Windows, pelo que os programadores que queiram desenvolver aplicações para esta plataforma a devem conhecer.

A linguagem C# é considerada uma linguagem de alto nível, ou seja, uma linguagem com um nível de abstração relativamente elevado longe do código de máquina e mais próxima da linguagem humana, não necessitando o programador de conhecer a arquitetura do processador (registos e instruções) para poder programar.

A sintaxe da linguagem C#, à semelhança de outras linguagens como o VB, C, C++, Java ou Python, utiliza termos comuns da língua inglesa, o que a torna, fácil de compreender e utilizar para quem saiba falar inglês.

Dada a semelhança da sintaxe do C# com as linguagens de programação acima referidas ao aprender a programar em C# está também aprender a programar nessas linguagens.

A linguagem C# segue o paradigma da programação orientada pelos objetos. Embora o presente livro se foque na parte procedimental/imperativa da linguagem vão aparecendo ao longo do livro algumas situações em que é necessário utilizar alguns elementos relacionados com este paradigma como a utilização dos operadores "." ou "*new*".

Dado ser uma linguagem orientada pelos objetos, os programas serão sempre criados dentro de uma classe, mais concretamente dentro da classe "*Program*", que está dentro de um *Namespace* com o mesmo nome do projeto, definido por defeito aquando da criação do projeto no *Visual Studio*. No último capítulo, em que é feita uma breve introdução ao paradigma da programação orientada pelos objetos, irá utilizar outras classes para implementar os exemplos.

3. *.NET Framework*

Os programas escritos na linguagem C# são executados na plataforma *.NET* (lê-se dotNet), uma plataforma de execução de aplicações no Windows. Esta plataforma inclui um sistema de execução virtual designado de CLR (*Common Language Runtime*) e uma extensa biblioteca de classes com funcionalidades que vão desde a apresentação dos dados no ecrã em modo consola, à manipulação de ficheiros e acesso a bases de dados.

O código escrito em C# é compilado (convertido) numa linguagem intermediária, designada comummente por MSIL (Microsoft Intermediate Language) que está em conformidade com a especificação da CLI (*Common Language Infrastructure*). O resultado final é uma ficheiro executável (extensão .exe ou .dll) na plataforma .NET que inclui para além do código fonte compilado os recursos associados à solução, como ficheiros de imagens, designado *Assembly*.

Quando o programa em C# é executado, o *Assembly* é carregado no CLR, que, caso as condições estejam satisfeitas, executará a compilação JIT (*Just In Time*) para converter o código MSIL em instruções nativas da máquina.

O diagrama da figura 1.1 mostra o processo que se acabou de descrever.

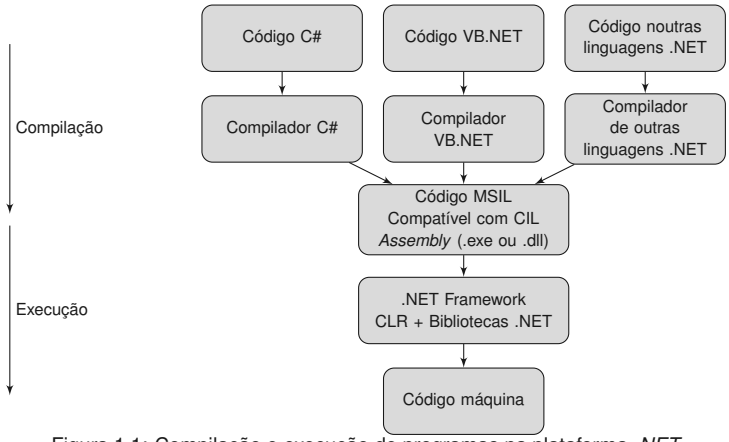

Figura 1.1: Compilação e execução de programas na plataforma *.NET*

⚠ Antes de se criar uma nova funcionalidade deve-se pesquisar nas bibliotecas da linguagem C# se não existe já um método ou classe para o que se deseja.

A Microsoft com a aquisição do produto Xamarin passou a disponibilizar aos seus programadores a possibilidade de desenvolverem aplicações na linguagem C# para sistemas operativos para além do Windows, como o Mac OS e os sistemas operativos móveis Android e iOS, ou seja, ao aprender a programar com C# irá ganhar competências para programar para diferentes dispositivos para além dos que corem o sistema operativo Windows.

A figura abaixo apresentada por Scott Hunter e Scott Hanselman sobre o futuro da plataforma *.NET* mostra que as intenções da Microsoft passam por desenvolver uma livraria comum - a "*.NET standard library*" - para o desenvolvimento de aplicações para os diferentes dispositivos/plataformas.

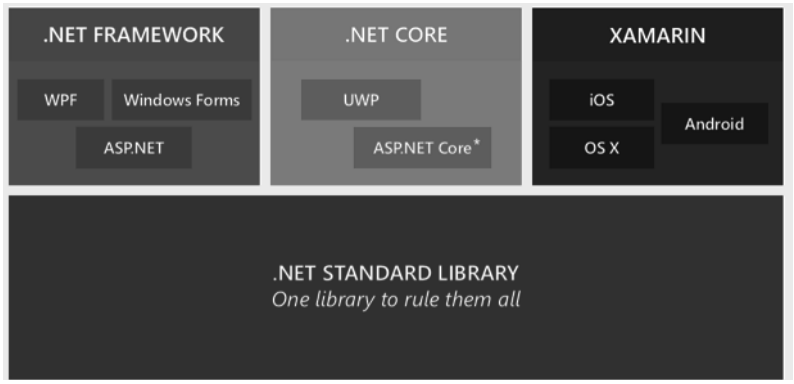

Figura 1.2: Futuro da plataforma ".NET".
Fonte: ".NET Overview", Scott Hunter e Scott Hanselman

Capítulo 2

Primeiros programas

Este capítulo é um capítulo de motivação em que irá instalar o Visual Studio (VS), criar os seus primeiros programas e ser introduzido a alguns elementos básicos da programação como variáveis, decisões e repetições. Nos capítulos seguintes irá consolidar os conceitos aqui apresentados.

1. Instalação do Visual Studio

A versão do VS utilizada neste livro é a Visual Studio Community 2017, uma versão gratuita cujos passos de instalação se descrevem de seguida.

O primeiro passo é aceder ao Google e pesquisar utilizando as palavras "Visual Studio" combinadas com as palavras "gratuito", "*free*", etc. Na lista de resultados desta pesquisa irá aparecer a edição *Community* que é aquela que interessa.

Após descobrir a hiperligação para o pacote de instalação (no momento da escrita do livro disponível em `https://www.visualstudio.com/pt-br/vs/community/`), descarregue o arquivo e dê início ao processo de instalação. Após iniciar o processo de instalação ir-lhe-á aparecer a janela apresentada na figura abaixo (ou similar, dependendo da versão do VS) onde deverá escolher pelo menos as opções "Desenvolvimento com a plataforma Universal do Windows" e "Desenvolvimento para Desktop com o .NET".

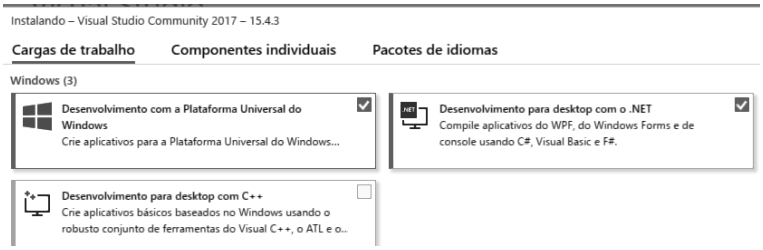

Figura 2.1: Instalação do VS 2017

Após a instalação do VS execute o mesmo. Caso não saiba onde está o VS pesquise por "Visual Studio" na pesquisa do Windows.

Da primeira vez que executa o programa ser-lhe-á solicitado a introdução das credenciais relativas à sua conta Microsoft para autenticação. Embora exista a opção de continuar sem criar uma conta Microsoft recomenda-se que crie uma, caso não possua, para usufruir de outros serviços Microsoft associados ao VS e à plataforma *.NET*. Após a autenticação feita, o programa inicia mostrando uma "Página inicial" com várias opções (ver figura 2.2). A opção que interessa para começar a fazer programas em C# é a opção "Criar novo projeto..." que se encontra na área "Novo projeto", destacada na figura 2.2.

Figura 2.2: Janela inicial do VS 2017

2. Criar um projeto

Para criar um novo projeto clica-se na ligação “Criar novo projeto...” da “Página inicial” (ver figura 2.2). Também se pode aceder a esta operação através do menu (Arquivo-> Novo -> Projeto).

Após iniciar a criação do projeto, o VS apresenta-lhe um formulário onde terá de escolher o tipo de aplicação a criar (ver figura 2.3). Neste formulário deverá primeiro escolher a linguagem em que quer codificar a aplicação, no caso deste livro o C#, e depois selecionar o tipo de aplicação a utilizar, neste caso, “Aplicativo do Console (.NET Framework)”.

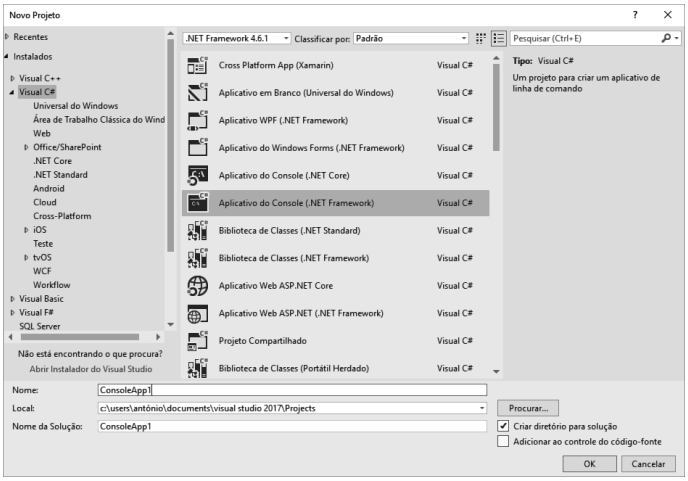

Figura 2.3: Criação de um novo projeto C# do tipo Consola

António Trigo é docente do ISCAC – Coimbra Business School, do Instituto Politécnico de Coimbra, onde leciona Algoritmos e Programação. É membro do grupo de investigação Information Systems and Technologies for Transformation of Organizations and Society (ISTTOS), do centro de investigação Algoritmi, da Universidade do Minho.

Jorge Henriques é docente do ISCAC – Coimbra Business School, do Instituto Politécnico de Coimbra, onde leciona Algoritmos e Programação.

Se nunca programou ou está interessado em aprender C# este livro é para si.

Não é necessário qualquer conhecimento prévio de uma linguagem de programação, somente estar à vontade com um computador.

Tendo como filosofia o aprender fazendo, este livro propõe uma abordagem de aprendizagem passo a passo em que os conceitos são acompanhados de exemplos práticos. O objetivo é que veja, logo desde o início, os resultados do seu estudo e assim se motive a completar a aprendizagem proposta neste livro. No fim de cada capítulo propõe-se um conjunto de exercícios para consolidar os conceitos apresentados, cujo código, à semelhança dos exemplos, está disponível para *download* na página da rede do livro.

Neste livro irá ter contacto com os principais conceitos da programação estruturada, o primeiro paradigma com o qual os estudantes de programação têm contacto, cobrindo os seguintes tópicos:

- Tipos de dados, variáveis, constantes e operadores.
- Leitura e escrita de dados (consola).
- Instruções de decisão.
- Instruções de repetição.
- Métodos (funções e procedimentos).
- Vetores e matrizes.
- Manipulação de texto.
- Tratamento de erros.

Termina com a implementação de um jogo com base nos conceitos transmitidos e uma breve introdução ao paradigma da programação orientada a objetos, em que assenta a linguagem C#.

ISBN 978-972-618-934-3

9 789726 189343

577