Richard Morgan

Cambridge IGCSE®
Computer Science
Programming Book
for Microsoft® Visual Basic

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

Information on this title: education.cambridge.org

First published 2015

Printed in the United Kingdom by Latimer Trend

A catalogue record for this publication is available from the British Library

ISBN 978-1-107-51864-3 Paperback

All examination-style questions, sample mark schemes, solutions and/or comments that appear in this book were written by the author. In examination, the way marks would be awarded to answers like these may be different.

Contents

Introduction

When I wrote this book I had two aims in mind. The first was to provide a programming book that specifically covered the material relevant to the Cambridge IGCSE® syllabus. The second, and perhaps more important, aim was to provide the student with a start to the exciting and rewarding process of being able to create their own computer programs.

Language

The syntax and structures used to implement programming techniques will vary across different languages. The book is entirely based around Visual Basic, one of the three recommended languages for the A Level syllabus. Visual Basic offers the student, as a programmer, two modes of application. There is a simple console window in which the student can learn and develop programming skills. It also offers a Windows Forms application, which allows the student to program commercial-style applications that offer a graphical user interface through which users can interact with programs.

The language is supported by a fully functional development environment called Visual Studio Express, which is available free directly from Microsoft. They also provide excellent support and language-specific tutorials via the Microsoft Developer Network. All the code and language specific comments in this book relate to Visual Studio Express 2013.

Examination focussed

The course will test computational thinking independent of any specific programming language. It will do this through the use of program design tools such as structure diagrams and flowcharts. It will also make use of pseudocode, a structured method for describing the logic of computer programs.

It is crucial that the student becomes familiar with these techniques. Throughout this book all the programming techniques are demonstrated in the non-language-specific format required. This will help prepare the student to answer the types of question they will meet in their studies.

To support learning, many of the chapters include examination-style tasks. Chapter 13 has examples of appropriate code solutions showing how to turn logical ideas into actual programs. There is also a series of examination-style questions in Chapter 12, which has a sample mark scheme giving possible solutions and showing where the marks might be awarded.

Developing programming skills

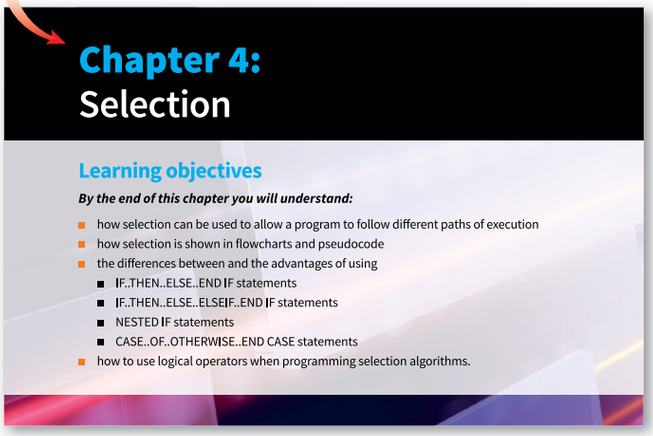
One of the advantages of Visual Basic is that it provides a language that encourages the student to program solutions making use of the basic programming constructs: sequence, selection and iteration. Although the language does have access to many powerful pre-written code libraries, they are not generally used in this book.

Computational thinking is the ability to resolve a problem into its constituent parts and to provide a logical and efficient coded solution. Experience tells me that knowing how to think computationally relies much more on an understanding of the underlying programming concepts than on the ability to learn a few shortcut library routines.

This book is aimed at teaching those underlying skills which can be applied to the languages of the future. It is without doubt that programming languages will develop over the coming years but the ability to think computationally will remain a constant.

How to use this book: a guided tour

Chapter – each chapter begins with a short list of the facts and concepts that are explained in it.



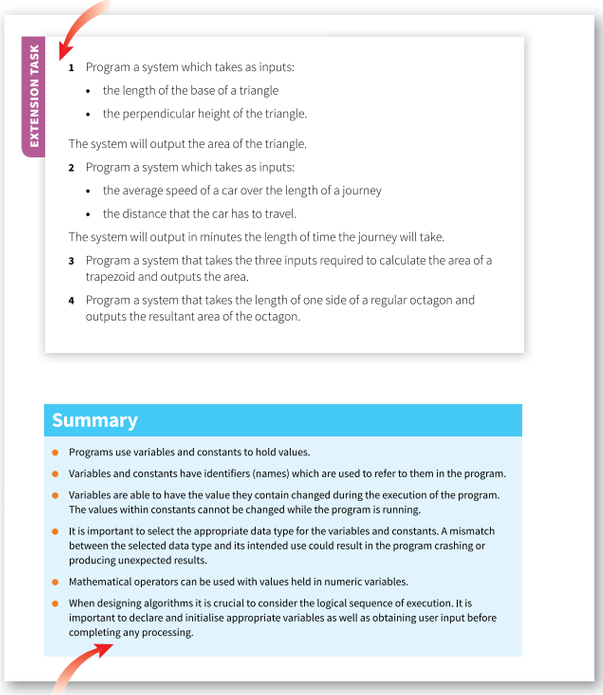
Chapter 4:
Selection

Learning objectives

By the end of this chapter you will understand:

- how selection can be used to allow a program to follow different paths of execution
- how selection is shown in flowcharts and pseudocode
- the differences between and the advantages of using
  - IF..THEN..ELSE..END IF statements
  - IF..THEN..ELSE..ELSEIF..END IF statements
  - NESTED IF statements
  - CASE..OF..OTHERWISE..END CASE statements
- how to use logical operators when programming selection algorithms.

Extension Task – extension of an existing exercise for the student to further develop their knowledge and understanding.



EXTENSION TASK

1 Program a system which takes as inputs:
- the length of the base of a triangle
- the perpendicular height of the triangle.

The system will output the area of the triangle.

2 Program a system which takes as inputs:
- the average speed of a car over the length of a journey
- the distance that the car has to travel.

The system will output in minutes the length of time the journey will take.

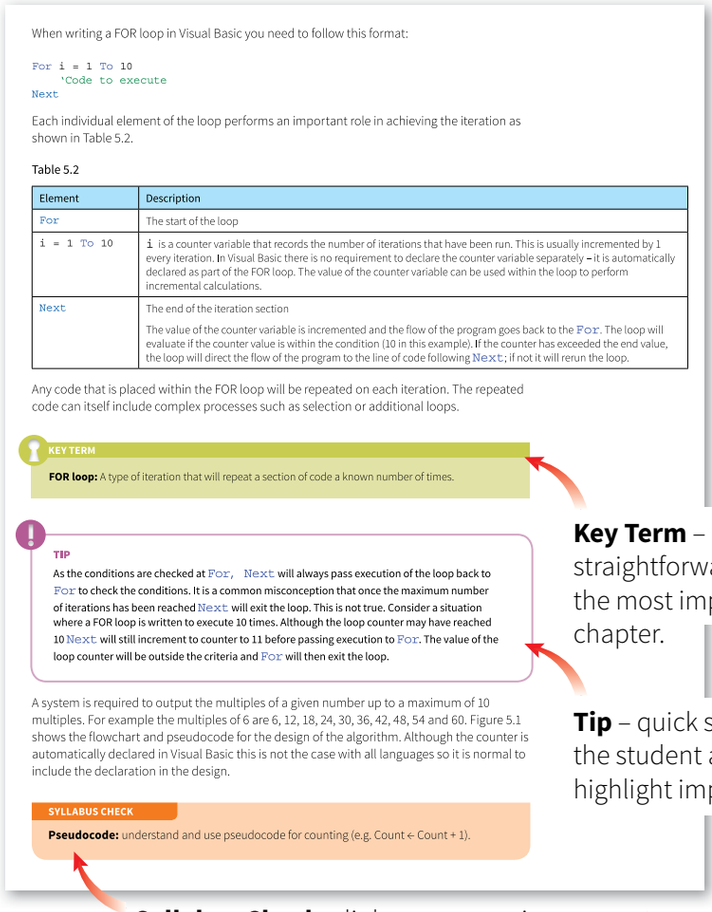
3 Program a system that takes the three inputs required to calculate the area of a trapezoid and outputs the area.

4 Program a system that takes the length of one side of a regular octagon and outputs the resultant area of the octagon.

Summary

- Programs use variables and constants to hold values.
- Variables and constants have identifiers (names) which are used to refer to them in the program.
- Variables are able to have the value they contain changed during the execution of the program. The values within constants cannot be changed while the program is running.
- It is important to select the appropriate data type for the variables and constants. A mismatch between the selected data type and its intended use could result in the program crashing or producing unexpected results.
- Mathematical operators can be used with values held in numeric variables.
- When designing algorithms it is crucial to consider the logical sequence of execution. It is important to declare and initialise appropriate variables as well as obtaining user input before completing any processing.

Summary Checklist – at the end of each chapter to review what the student has learned.



When writing a FOR loop in Visual Basic you need to follow this format:

```
For i = 1 To 10
    'Code to execute
Next
```

Each individual element of the loop performs an important role in achieving the iteration as shown in Table 5.2.

Table 5.2

Element	Description
`For`	The start of the loop
`i = 1 To 10`	`i` is a counter variable that records the number of iterations that have been run. This is usually incremented by 1 every iteration. In Visual Basic there is no requirement to declare the counter variable separately – it is automatically declared as part of the FOR loop. The value of the counter variable can be used within the loop to perform incremental calculations.
`Next`	The end of the iteration section The value of the counter variable is incremented and the flow of the program goes back to the `For`. The loop will evaluate if the counter value is within the condition (10 in this example). If the counter has exceeded the end value, the loop will direct the flow of the program to the line of code following `Next`; if not it will rerun the loop.

Any code that is placed within the FOR loop will be repeated on each iteration. The repeated code can itself include complex processes such as selection or additional loops.

KEY TERM

FOR loop: A type of iteration that will repeat a section of code a known number of times.

TIP

As the conditions are checked at `For`, `Next` will always pass execution of the loop back to `For` to check the conditions. It is a common misconception that once the maximum number of iterations has been reached `Next` will exit the loop. This is not true. Consider a situation where a FOR loop is written to execute 10 times. Although the loop counter may have reached 10 `Next` will still increment to counter to 11 before passing execution to `For`. The value of the loop counter will be outside the criteria and `For` will then exit the loop.

A system is required to output the multiples of a given number up to a maximum of 10 multiples. For example the multiples of 6 are 6, 12, 18, 24, 30, 36, 42, 48, 54 and 60. Figure 5.1 shows the flowchart and pseudocode for the design of the algorithm. Although the counter is automatically declared in Visual Basic this is not the case with all languages so it is normal to include the declaration in the design.

SYLLABUS CHECK

Pseudocode: understand and use pseudocode for counting (e.g. Count ← Count + 1).

TASK

FOR Loop

1 Extend the multiply system to include two inputs. The first input is the number to multiply, the second is the number of multiples required.

2 Produce a system that accepts two numbers A and B and outputs A^B. For example if A = 3 and B = 4, the output will be 81 ($A^4 = A \times A \times A \times A$).

Task – exercises for the student to test their knowledge and understanding.

Key Term – clear and straightforward explanations of the most important terms in each chapter.

Tip – quick suggestions to remind the student about key facts and highlight important points.

Syllabus Check – links programming concepts explained in the text to the Cambridge IGCSE syllabus.

Acknowledgements

The authors and publishers acknowledge the following sources of copyright material and are grateful for the permissions granted.

Cover Soulart/Shutterstock; p. 1 isak55/Shutterstock; p. 13 aimy27feb/Shutterstock; p. 19 Image Source/Getty Images; p. 31 Magictorch/Ikon Images/Getty Images; p. 51 alexaldo/iStock/Getty Images; p. 67 Ioana Davies (Drutu)/Shutterstock; p. 77 Devrimb/iStock/Getty Images; p. 85 Mclek/Shutterstock; p. 93 Kutay Tanir/Photodisc/Getty Images; p. 103 ILeysen/Shutterstock; p. 113 Kamil Krawczyk/E+/Getty Images; p. 114 John Howard/Science Photo Library

Screenshots of Microsoft Visual Studio Express 2013 for Windows used with permission from Microsoft.

Cambridge IGCSE® Computer Science Programming Book is an independent publication and is not affiliated with, nor has it been authorized, sponsored, or otherwise approved by Microsoft Corporation.

The publisher has used its best endeavours to ensure that the URLs for external websites referred to in this product are correct and active at the time of going to press. However, the publisher has no responsibility for external websites and can make no guarantee that a site will remain live or that the content is or will remain appropriate.