

# SYLLABUS

**Cambridge O Level  
Computer Science**

**2210**

For examination in June and November 2015

## Changes to syllabus for 2015

- Change of syllabus name:  
Previously named '7010 Computer Studies', this syllabus has been reviewed throughout to bring it up to date and to allow learners to begin the development of their computational thinking and programming skills. Teachers are advised to read the whole syllabus document before planning their teaching programmes. As 'Computer Science', this syllabus now shares the same name as the AS/AL syllabus (formerly AS/AL Computing), indicating the firm links and progression between these syllabuses.
- New syllabus code: 2210
- Some changes to the assessment structure, but with many features retained (see Section 4):
  - all components are externally assessed.
  - Paper 1 Theory is now 1 hour 45 minutes; 60% weighting; 75 marks.
  - New Paper 2 Problem-solving and Programming: 1 hour 45 minutes; 40% weighting; 50 marks. This paper replaces Paper 2 Coursework and Paper 3 Alternative to Coursework. There are pre-release materials for Paper 2 Problem-solving and Programming for learners to complete practical tasks. Teachers are expected to incorporate the pre-release material tasks into their lessons and give support in finding methods and reaching solutions.
- There is no coursework.
- Clarified syllabus aims and assessment objectives (see Section 5)
- Revised syllabus content (see Section 6):
  - one new practical topic – introducing candidates to the concept of arrays and so enabling learners to develop programming solutions for real-world problems
  - two new theory topics: computer ethics and hexadecimal numbers
  - one topic removed: systems life cycle.

Cambridge International Examinations retains the copyright on all its publications. Registered Centres are permitted to copy material from this booklet for their own internal use. However, we cannot give permission to Centres to photocopy any material that is acknowledged to a third party even for internal use within a Centre.

® IGCSE is the registered trademark of Cambridge International Examinations

© Cambridge International Examinations 2013

---

# Contents

---

1. Introduction .....	2
1.1 Why choose Cambridge?	
1.2 Why choose Cambridge O Level?	
1.3 Why choose Cambridge O Level Computer Science?	
1.4 How can I find out more?	
2. Teacher support.....	5
2.1 Support materials	
2.2 Resource lists	
2.3 Training	
3. Syllabus content at a glance .....	6
4. Assessment at a glance .....	7
5. Syllabus aims and assessment objectives .....	8
5.1 Syllabus aims	
5.2 Assessment objectives	
5.3 Relationship between assessment objectives and components	
6. Syllabus content .....	9
7. Description of components .....	16
7.1 Scheme of assessment	
7.2 Paper 2 Problem-solving and Programming pre-release materials	
8. Notes for the guidance of teachers .....	17
9. Other information .....	18

---

# 1. Introduction

---

## 1.1 Why choose Cambridge?

### Recognition

Cambridge International Examinations is the world's largest provider of international education programmes and qualifications for learners aged 5 to 19. We are part of Cambridge Assessment, a department of the University of Cambridge, trusted for excellence in education. Our qualifications are recognised by the world's universities and employers.

Cambridge O Level is internationally recognised by schools, universities and employers as equivalent in demand to Cambridge IGCSE® (International General Certificate of Secondary Education). Learn more at [www.cie.org.uk/recognition](http://www.cie.org.uk/recognition)

### Excellence in education

Our mission is to deliver world-class international education through the provision of high-quality curricula, assessment and services.

More than 9000 schools are part of our Cambridge learning community. We support teachers in over 160 countries who offer their learners an international education based on our curricula and leading to our qualifications. Every year, thousands of learners use Cambridge qualifications to gain places at universities around the world.

Our syllabuses are reviewed and updated regularly so that they reflect the latest thinking of international experts and practitioners and take account of the different national contexts in which they are taught.

Cambridge programmes and qualifications are designed to support learners in becoming:

- **confident** in working with information and ideas – their own and those of others
- **responsible** for themselves, responsive to and respectful of others
- **reflective** as learners, developing their ability to learn
- **innovative** and equipped for new and future challenges
- **engaged** intellectually and socially, ready to make a difference.

### Support for teachers

A wide range of materials and resources is available to support teachers and learners in Cambridge schools. Resources suit a variety of teaching methods in different international contexts. Through subject discussion forums and training, teachers can access the expert advice they need for teaching our qualifications. More details can be found in Section 2 of this syllabus and at [www.cie.org.uk/teachers](http://www.cie.org.uk/teachers)

### Support for exams officers

Exams officers can trust in reliable, efficient administration of exams entries and excellent personal support from our customer services. Learn more at [www.cie.org.uk/examsOfficers](http://www.cie.org.uk/examsOfficers)

## Not-for-profit, part of the University of Cambridge

We are a not-for-profit organisation where the needs of the teachers and learners are at the core of what we do. We continually invest in educational research and respond to feedback from our customers in order to improve our qualifications, products and services.

Our systems for managing the provision of international qualifications and education programmes for learners aged 5 to 19 are certified as meeting the internationally recognised standard for quality management, ISO 9001:2008. Learn more at [www.cie.org.uk/ISO9001](http://www.cie.org.uk/ISO9001)

## 1.2 Why choose Cambridge O Level?

Cambridge O Levels have been designed for an international audience and are sensitive to the needs of different countries. These qualifications are designed for learners whose first language may not be English and this is acknowledged throughout the examination process. The Cambridge O Level syllabus also allows teaching to be placed in a localised context, making it relevant in varying regions.

Our aim is to balance knowledge, understanding and skills in our programmes and qualifications to enable candidates to become effective learners and to provide a solid foundation for their continuing educational journey.

Through our professional development courses and our support materials for Cambridge O Levels, we provide the tools to enable teachers to prepare learners to the best of their ability and work with us in the pursuit of excellence in education.

Cambridge O Levels are considered to be an excellent preparation for Cambridge International AS and A Levels, the Cambridge AICE (Advanced International Certificate of Education) Group Award, Cambridge Pre-U, and other education programmes, such as the US Advanced Placement program and the International Baccalaureate Diploma programme. Learn more about Cambridge O Levels at [www.cie.org.uk/cambridgesecundary2](http://www.cie.org.uk/cambridgesecundary2)

## Guided learning hours

Cambridge O Level syllabuses are designed on the assumption that learners have about 130 guided learning hours per subject over the duration of the course, but this is for guidance only. The number of hours required to gain the qualification may vary according to local curricular practice and the learners' prior experience of the subject.

## 1.3 Why choose Cambridge O Level Computer Science?

The Cambridge O Level Computer Science syllabus enables learners to develop an interest in computing and gain confidence in computational thinking and programming. Cambridge O Level Computer Science is an ideal foundation for further study at Cambridge International A Level, and the skills learnt can also be used in other areas of study and in everyday life.

## Prior learning

Candidates beginning this course are not expected to have studied computer science or IT previously.

## Progression

Cambridge O Level Certificates are general qualifications that enable candidates either to progress directly to employment, or to proceed to further qualifications.

Candidates who are awarded grades A\* to C in Cambridge O Level Computer Science are well prepared to follow courses leading to Cambridge International AS and A Level Computer Science, or the equivalent.

### 1.4 How can I find out more?

#### If you are already a Cambridge school

You can make entries for this qualification through your usual channels. If you have any questions, please contact us at **info@cie.org.uk**

#### If you are not yet a Cambridge school

Learn about the benefits of becoming a Cambridge school at **www.cie.org.uk/startcambridge**. Email us at **info@cie.org.uk** to find out how your organisation can register to become a Cambridge school.

---

## 2. Teacher support

---

### 2.1 Support materials

Cambridge syllabuses, past question papers and examiner reports to cover the last examination series are on the *Syllabus and Support Materials* DVD, which we send to all Cambridge schools.

You can also go to our public website at [www.cie.org.uk/olevel](http://www.cie.org.uk/olevel) to download current and future syllabuses together with specimen papers or past question papers and examiner reports from one series.

For teachers at registered Cambridge schools a range of additional support materials for specific syllabuses is available online. For Teacher Support go to <http://teachers.cie.org.uk> (username and password required).

### 2.2 Resource lists

We work with publishers providing a range of resources for our syllabuses including textbooks, websites, CDs etc. Any endorsed, recommended and suggested resources are listed on both our public website and on Teacher Support.

The resource lists can be filtered to show all resources or just those which are endorsed or recommended by Cambridge. Resources endorsed by Cambridge go through a detailed quality assurance process and are written to align closely with the Cambridge syllabus they support.

### 2.3 Training

We offer a range of support activities for teachers to ensure they have the relevant knowledge and skills to deliver our qualifications. See [www.cie.org.uk/events](http://www.cie.org.uk/events) for further information.

### 3. Syllabus content at a glance

All candidates study the following topics.

Sections	Topics
<b>Section 1</b> <b>Theory of Computer Science</b>	1.1 Data representation <ul style="list-style-type: none"> <li>1.1.1 Binary systems</li> <li>1.1.2 Hexadecimal</li> <li>1.1.3 Data storage</li> </ul> 1.2 Communication and Internet technologies <ul style="list-style-type: none"> <li>1.2.1 Serial and parallel data transmission</li> <li>1.2.2 Security aspects</li> <li>1.2.3 Internet principles of operation</li> </ul> 1.3 Hardware and software <ul style="list-style-type: none"> <li>1.3.1 Logic gates</li> <li>1.3.2 Computer architecture and the fetch-execute cycle</li> <li>1.3.3 Input devices</li> <li>1.3.4 Output devices</li> <li>1.3.5 Memory, storage devices and media</li> <li>1.3.6 Operating systems</li> <li>1.3.7 High- and low-level languages and their translators</li> </ul> 1.4 Security 1.5 Ethics
<b>Section 2</b> <b>Practical Problem-solving and Programming</b>	2.1 Algorithm design and problem-solving <ul style="list-style-type: none"> <li>2.1.1 Problem-solving and design</li> <li>2.1.2 Pseudocode</li> </ul> 2.2 Programming <ul style="list-style-type: none"> <li>2.2.1 Programming concepts</li> <li>2.2.2 Data structures; arrays</li> </ul> 2.3 Databases



## 4. Assessment at a glance

For Cambridge O Level Computer Science, candidates take two components: Paper 1 and Paper 2.

Components		Weighting
<p><b>Paper 1: Theory</b></p> <p>This written paper contains short-answer and structured questions. There is no choice of questions.</p> <p>No calculators are permitted in this paper.</p> <p>75 marks</p> <p>Externally assessed.</p>	1 hour 45 minutes	60%
<p><b>Paper 2: Problem-solving and Programming</b></p> <p>This written paper contains short-answer and structured questions. There is no choice of questions. 20 of the marks for this paper are from questions set on the pre-release material.<sup>1</sup></p> <p>No calculators are permitted in this paper.</p> <p>50 marks</p> <p>Externally assessed.</p>	1 hour 45 minutes	40%

### Availability

This syllabus is examined in the June and November examination series.

This syllabus is available to private candidates.

Detailed timetables are available from [www.cie.org.uk/examsOfficers](http://www.cie.org.uk/examsOfficers)

Centres in the UK that receive government funding are advised to consult the Cambridge website [www.cie.org.uk](http://www.cie.org.uk) for the latest information before beginning to teach this syllabus.

### Combining this with other syllabuses

Candidates can combine this syllabus in an examination series with any other Cambridge syllabus, except:

- syllabuses with the same title at the same level
- 0478 Cambridge IGCSE Computer Science

Please note that Cambridge O Level, Cambridge IGCSE and Cambridge International Level 1/Level 2 Certificate syllabuses are at the same level.

<sup>1</sup> The pre-release material will be made available to Centres six months before the examination. It will also be reproduced in the question paper. Candidates are not permitted to bring any prepared material into the examination.

## 5. Syllabus aims and assessment objectives

### 5.1 Syllabus aims

The Cambridge O Level Computer Science syllabus aims are to:

- develop computational thinking
- develop an understanding of the main principles of solving problems using computers
- develop understanding that every computer system is made up of sub-systems, which in turn consist of further sub-systems
- develop an understanding of the component parts of computer systems and how they interrelate, including software, data, hardware, communications and people
- acquire the skills necessary to apply this understanding to develop computer-based solutions to problems using a high-level programming language.

Computer science is the study of the foundational principles and practices of computation and computational thinking and their application in the design and development of computer systems.

This syllabus aims to encourage candidates to develop computational thinking, that is thinking about what can be computed and how, and includes consideration of the data required. Learning computational thinking involves learning to program, that is to write computer code, because this is the means by which computational thinking is expressed.

The assessment is by written papers, but the learning should be done in a mainly practical way: problem-solving and programming. Questions will require the candidate to think, use knowledge with understanding and demonstrate understanding gained through practising practical skills. Questions will not revolve around pure recall.

### 5.2 Assessment objectives

**AO1** Recall, select and communicate knowledge and understanding of computer technology.

**AO2** Apply knowledge, understanding and skills to solve computing or programming problems.

**AO3** Analyse, evaluate, make reasoned judgements and present conclusions.

### 5.3 Relationship between assessment objectives and components

The **approximate** weightings allocated to each of the assessment objectives are summarised below.

Assessment objective	Paper 1	Paper 2	Weighting for qualification
AO1	32%	8%	40%
AO2	16%	24%	40%
AO3	12%	8%	20%
Total	60%	40%	100%

---

## 6. Syllabus content

---

### Section 1 Theory of Computer Science

#### 1.1 Data representation

Candidates should be able to:

##### 1.1.1 Binary systems

- recognise the use of binary numbers in computer systems
- convert denary numbers into binary and binary numbers into denary
- show understanding of the concept of a byte and how the byte is used to measure memory size
- use binary in computer registers for a given application (such as in robotics, digital instruments and counting systems)

##### 1.1.2 Hexadecimal

- represent integers as hexadecimal numbers
- show understanding of the reasons for choosing hexadecimal to represent numbers
- convert positive hexadecimal integers to and from denary
- convert positive hexadecimal integers to and from binary
- represent numbers stored in registers and main memory as hexadecimal
- identify current uses of hexadecimal numbers in computing, such as defining colours in Hypertext Markup Language (HTML), Media Access Control (MAC) addresses, assembly languages and machine code, debugging

##### 1.1.3 Data storage

- show understanding that sound (music), pictures, video, text and numbers are stored in different formats
- identify and describe methods of error detection and correction, such as parity checks, check digits, checksums and Automatic Repeat reQuests (ARQ)
- show understanding of the concept of Musical Instrument Digital Interface (MIDI) files, jpeg files, MP3 and MP4 files
- show understanding of the principles of data compression (lossless and lossy compression algorithms) applied to music/video, photos and text files

## 1.2 Communication and Internet technologies

Candidates should be able to:

### 1.2.1 Serial and parallel data transmission

- show understanding of what is meant by transmission of data
- distinguish between serial and parallel data transmission
- show understanding of the reasons for choosing serial or parallel data transmission
- show understanding of the need to check for errors
- explain how parity bits are used for error detection
- identify current uses of serial and parallel data transmission, such as Integrated Circuits (IC) and Universal Serial Bus (USB)

### 1.2.2 Security aspects

- show understanding of the security aspects of using the Internet and understand what methods are available to help minimise the risks
- show understanding of the Internet risks associated with malware, including viruses, spyware and hacking
- explain how anti-virus and other protection software helps to protect the user from security risks (this also links into section 1.4 of the syllabus)

### 1.2.3 Internet principles of operation

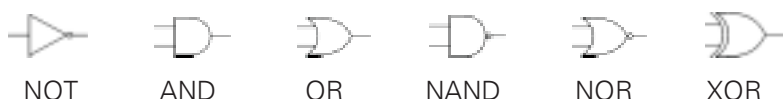
- show understanding of the role of the browser and Internet server
- show understanding of what is meant by hypertext transfer protocol (http) and HTML
- distinguish between HTML structure and presentation
- show understanding of the concept of MAC address, Internet Protocol (IP) address and cookies

### 1.3 Hardware and software

Candidates should be able to:

#### 1.3.1 Logic gates

- use logic gates to create electronic circuits
- understand and define the functions of NOT, AND, OR, NAND, NOR and XOR (EOR) gates, including the binary output produced from all the possible binary inputs (all gates, except the NOT gate, will have 2 inputs only)
- draw truth tables and recognise a logic gate from its truth table
- recognise and use the following standard symbols used to represent logic gates:



- produce truth tables for given logic circuits, for example:

<b>a</b>	<b>b</b>	<b>c</b>	<b>out</b>
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

- produce a logic circuit to solve a given problem or to implement a given written logic statement, such as IF (switch A is NOT on) OR (switch B is on AND switch C is NOT on) then alarm, X, sounds

#### 1.3.2 Computer architecture and the fetch-execute cycle

- show understanding of the basic Von Neumann model for a computer system and the stored program concept (program instructions and data are stored in main memory and instructions are fetched and executed one after another)
- describe the stages of the fetch-execute cycle

#### 1.3.3 Input devices

- describe the principles of operation (how each device works) of a range of input devices including 2D and 3D scanners, barcode readers, digital cameras, keyboards, mice, touch screens, microphones
- describe how these principles are applied to real-life scenarios, for example: scanning of passports at airports, barcode readers at supermarket checkouts, and touch screens on mobile devices
- describe how a range of sensors can be used to input data into a computer system, including light, temperature, magnetic field, gas, pressure, moisture, humidity, pH/acidity/alkalinity and motion/infra-red
- describe how these sensors are used in real-life scenarios, for example: street lights, security devices, pollution control, games, and household and industrial applications

#### 1.3.4 Output devices

- describe the principles of operation of a range of output devices, including: inkjet, laser and 3D printers; 2D and 3D cutters; speakers and headphones; actuators; flat-panel display screens, including Liquid Crystal Display (LCD) and Light-Emitting Diodes (LED); and LCD projectors and Digital Light Projectors (DLP)
- describe how these principles are applied to real-life scenarios for example: printing single items on demand or in large volumes; banks of digital displays; use of small screens on mobile devices; smart boards

#### 1.3.5 Memory, storage devices and media

- show understanding of the difference between: primary, secondary and off-line storage and provide examples of each, such as, primary: Read Only Memory (ROM), Random Access Memory (RAM) and DVD-RAM; secondary: hard disk drive (HDD) and Solid State Drives (SSDs); off-line: Digital Versatile Disks (DVDs), Compact Disks (CDs), Blu-ray, USB flash memory and removable disks
- describe the principles of operation of a range of types of storage devices and media including magnetic, optical and solid state
- describe how these principles are applied to currently available storage solutions, such as SSDs, hard disk drives, USB flash memory, DVDs, CDs and Blu-ray
- calculate the storage requirement of a file

#### 1.3.6 Operating systems

- describe the purpose of an operating system
- show understanding of the need for interrupts

#### 1.3.7 High- and low-level languages and their translators

- show understanding of the need for both high-level and low-level languages
- show understanding of the need for compilers when translating programs written in a high-level language
- show understanding of the use of interpreters with high-level language programs
- show understanding of the need for assemblers when translating programs written in assembly language

## 1.4 Security

Candidates should be able to:

### 1.4.1

- show understanding of the need to keep data safe from accidental damage, including corruption and human errors
- show understanding of the need to keep data safe from malicious actions, including unauthorised viewing, deleting, copying and corruption

### 1.4.2

- show understanding of how data are kept safe when stored and transmitted, including:
  - use of passwords, both entered at a keyboard and biometric
  - use of firewalls, both software and hardware, including proxy servers
  - use of Secure Socket Layer (SSL)
  - use of symmetric encryption (plain text, cypher text and use of a key) showing understanding that increasing the length of a key increases the strength of the encryption

### 1.4.3

- show understanding of the need to keep online systems safe from attacks including denial of service attacks, phishing, pharming

### 1.4.4

- describe how the knowledge from 1.4.1, 1.4.2 and 1.4.3 can be applied to real-life scenarios including, for example, online banking, shopping

## 1.5 Ethics

Candidates should be able to:

- show understanding of computer ethics, including copyright issues and plagiarism
- distinguish between free software, freeware and shareware
- show understanding of the ethical issues raised by the spread of electronic communication and computer systems, including hacking, cracking and production of malware

## Section 2 Practical Problem-solving and Programming

### 2.1 Algorithm design and problem-solving

Candidates should be able to:

#### 2.1.1 Problem-solving and design

- show understanding that every computer system is made up of sub-systems, which in turn are made up of further sub-systems
- use top-down design, structure diagrams, flowcharts, pseudocode, library routines and sub-routines
- work out the purpose of a given algorithm
- explain standard methods of solution
- suggest and apply suitable test data
- understand the need for validation and verification checks to be made on input data (validation could include range checks, length checks, type checks and check digits)
- use trace tables to find the value of variables at each step in an algorithm
- identify errors in given algorithms and suggest ways of removing these errors
- produce an algorithm for a given problem (either in the form of pseudocode or flowchart)
- comment on the effectiveness of a given solution

#### 2.1.2 Pseudocode

- understand and use pseudocode for assignment, using  $\leftarrow$
- understand and use pseudocode, using the following conditional statements:  
`IF ... THEN ... ELSE ... ENDIF`  
`CASE ... OF ... OTHERWISE ... ENDCASE`
- understand and use pseudocode, using the following loop structures:  
`FOR ... TO ... NEXT`  
`REPEAT ... UNTIL`  
`WHILE ... DO ... ENDWHILE`
- understand and use pseudocode, using the following commands and statements:  
`INPUT and OUTPUT (e.g. READ and PRINT)`  
 totalling (e.g. `Sum  $\leftarrow$  Sum + Number`)  
 counting (e.g. `Count  $\leftarrow$  Count + 1`)

(Candidates are advised to try out solutions to a variety of different problems on a computer using a language of their choice; no particular programming language will be assumed in this syllabus.)



## 2.2 Programming

Candidates should be able to:

### 2.2.1 Programming concepts

- declare and use variables and constants
- understand and use basic data types: Integer, Real, Char, String and Boolean
- understand and use the concepts of sequence, selection, repetition, totalling and counting
- use predefined procedures/functions

### 2.2.2 Data structures; arrays

- declare the size of one-dimensional arrays, for example: A[1:n]
- show understanding of the use of a variable as an index in an array
- read values into an array using a FOR ... TO ... NEXT loop

## 2.3 Databases

Candidates should be able to:

- define a single-table database from given data storage requirements
- choose a suitable primary key for a database table
- perform a query-by-example from given search criteria

---

## 7. Description of components

---

### 7.1 Scheme of assessment

All candidates need to demonstrate basic levels of knowledge and understanding. To do this, they need to recall information and either apply that information or explain it. All candidates also need to demonstrate some level of problem-solving and practical skills. This will require them to show their ability to comprehend and analyse computational tasks and design, implement, test and evaluate solutions using a variety of methods.

#### Component 1: Paper 1 Theory

This is a compulsory question paper, consisting of short-answer and structured questions set on Section 1 of the syllabus. Candidates must answer all questions. Candidates will answer on the question paper.

#### Component 2: Paper 2 Problem-solving and Programming

This is a compulsory question paper, consisting of short-answer and structured questions set on Section 2 of the syllabus. Candidates must answer all questions. Candidates will answer on the question paper. 20 of the marks in this paper are from questions set on tasks provided in the Paper 2 Problem-solving and Programming pre-release materials.

Centres should spend approximately 50% of the total time studying this section of the syllabus. Centres should ensure that candidates are taught the contents of this section in a largely practical way. Candidates will be expected to be able to program in a high-level programming language to be chosen by the Centre. The programming language should be procedural. There will be some examining of knowledge with understanding, but the bulk of the credit will be for using techniques and skills to solve problems. In all cases the logic will be of more importance than the syntax.

### 7.2 Paper 2 Problem-solving and Programming pre-release materials

The Paper 2 Problem-solving and Programming pre-release materials will be made available to Centres six months before the exam. Centres are advised to encourage their candidates to develop solutions to tasks using a high-level programming language, such as Visual Basic, Pascal/Delphi or Python. The purpose of the pre-release material tasks is to direct candidates to some of the topics which will be examined in Paper 2. Teachers are expected to incorporate these tasks into their lessons and give support in finding methods and reaching solutions. 20 of the marks in this paper will be from questions testing candidates' understanding gained from developing programmed solutions to these tasks. The tasks will be appropriate for all ability levels.

The exam questions will require candidates to have practical programming experience including writing their own programs, executing (running), testing and debugging them. Knowledge of programming language syntax will not be examined. The higher ability candidates are to be encouraged to extend their practical programming beyond the scope of these tasks.

---

## 8. Notes for the guidance of teachers

---

### Introduction

The purpose of these notes is to provide assistance for teachers preparing candidates for the Cambridge O Level Computer Science examination. They contain notes on equipment, facilities and resources and sources of further information.

### Equipment and facilities

Computer science is a practical subject and the Cambridge O Level syllabus places emphasis on the use of procedural high-level programming languages. Centres must ensure that their equipment and facilities are adequate for candidates to be able to satisfy the requirements of the syllabus. The hardware facilities needed will depend on the number of candidates, but should be sufficient for all candidates to have enough time to practise their programming skills.

### Hardware

Candidates need to have access to a system with direct-access file capability on backing store and hardcopy facilities.

### Software

Candidates should have experience of using a high-level programming language, such as Visual Basic, Pascal/Delphi or Python, chosen by the Centre.

### Books

Provision of textbooks is difficult as new titles are available all the time. The British Computer Society (BCS) book list for schools and colleges lists books which are suitable for use as reference books. Teachers will need to consult several books to cover the whole syllabus adequately. There is a suggested book list on our website. Many schools prefer to have a wide range of reference books rather than a class textbook.

### Practical skills

Computer science is a practical subject and a range of practical exercises should supplement the study of most parts of the syllabus.

It is important that Centres encourage candidates, as early as possible in the course, to develop a systematic approach to practical problem-solving using appropriate resources.

---

## 9. Other information

---

### Equality and inclusion

Cambridge International Examinations has taken great care in the preparation of this syllabus and assessment materials to avoid bias of any kind. To comply with the UK Equality Act (2010), Cambridge has designed this qualification with the aim of avoiding direct and indirect discrimination.

The standard assessment arrangements may present unnecessary barriers for candidates with disabilities or learning difficulties. Arrangements can be put in place for these candidates to enable them to access the assessments and receive recognition of their attainment. Access arrangements will not be agreed if they give candidates an unfair advantage over others or if they compromise the standards being assessed.

Candidates who are unable to access the assessment of any component may be eligible to receive an award based on the parts of the assessment they have taken.

Information on access arrangements is found in the *Cambridge Handbook* which can be downloaded from the website **[www.cie.org.uk](http://www.cie.org.uk)**

### Language

This syllabus and the associated assessment materials are available in English only.

### Grading and reporting

Cambridge O Level results are shown by one of the grades A\*, A, B, C, D or E, indicating the standard achieved, A\* being the highest and E the lowest. 'Ungraded' indicates that the candidate's performance fell short of the standard required for grade E. 'Ungraded' will be reported on the statement of results but not on the certificate. The letters Q (result pending); X (no results) and Y (to be issued) may also appear on the statement of results but not on the certificate.

### Entry codes

To maintain the security of our examinations we produce question papers for different areas of the world, known as 'administrative zones'. Where the component entry code has two digits, the first digit is the component number given in the syllabus. The second digit is the location code, specific to an administrative zone. Information about entry codes, examination timetables and administrative instructions can be found in the *Cambridge Guide to Making Entries*.

Cambridge International Examinations  
1 Hills Road, Cambridge, CB1 2EU, United Kingdom  
Tel: +44 (0)1223 553554 Fax: +44 (0)1223 553558  
Email: [info@cie.org.uk](mailto:info@cie.org.uk) [www.cie.org.uk](http://www.cie.org.uk)

® IGCSE is the registered trademark of Cambridge International Examinations

© Cambridge International Examinations 2013

