MY REVISION NOTES

HODDER EDUCATION

OCR

GCSE (9–1)

COMPUTER SCIENCE

3RD EDITION

- ✚ Plan and organise your revision
- ✚ Reinforce skills and understanding
- ✚ Practise exam-style questions

For the J277 specification

MY REVISION NOTES
OCR GCSE (9–1)
COMPUTER SCIENCE

George Rouse

Boost

HODDER EDUCATION
LEARN MORE

This book draws on material written for and published in OCR GCSE Computer Science, Second Edition (978 1 5104 8416 0) by George Rouse, Lorne Pearcey and Gavin Craddock. The publisher would like to thank Lorne Pearcey and Gavin Craddock for permission to re-use their work in the present volume.

Lorne Pearcey and Gavin Craddock have not written any content specifically for this revision guide, including the exam-style questions and examiner's tips.

The Publishers would like to thank the following for permission to reproduce copyright material.

Photo credits
Figure 1.5.1: Background photograph © Mike Berenson/Colorado Captures/Getty Images

Acknowledgements
Adobe is either a registered trademark or trademark of Adobe in the United States and/or other countries.

Google and the Google logo are registered trademarks of Google LLC, used with permission.

Microsoft product screenshot(s) used with permission from Microsoft.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Every effort has been made to trace all copyright holders, but if any have been inadvertently overlooked, the Publishers will be pleased to make the necessary arrangements at the first opportunity.

Although every effort has been made to ensure that website addresses are correct at time of going to press, Hodder Education cannot be held responsible for the content of any website mentioned in this book. It is sometimes possible to find a relocated web page by typing in the address of the home page for a website in the URL window of your browser.

Hachette UK's policy is to use papers that are natural, renewable and recyclable products and made from wood grown in well-managed forests and other controlled sources. The logging and manufacturing processes are expected to conform to the environmental regulations of the country of origin.

Orders: please contact Hachette UK Distribution, Hely Hutchinson Centre, Milton Road, Didcot, Oxfordshire, OX11 7HH. Telephone: +44 (0)1235 827827. Email education@hachette.co.uk Lines are open from 9 a.m. to 5 p.m., Monday to Friday. You can also order through our website: www.hoddereducation.co.uk

ISBN: 978 1 3983 2114 4

First published in 2021 by
Hodder Education,
An Hachette UK Company
Carmelite House
50 Victoria Embankment
London EC4Y 0DZ

www.hoddereducation.co.uk

Impression number 10 9 8 7 6 5 4 3 2 1

Year 2025 2024 2023 2022 2021

Cover photo © Patrick P. Palej - stock.adobe.com

Illustrations by Aptara, Inc. and Integra Software Services Pvt. Ltd

Typeset in India by Aptara, Inc.

Printed in India

A catalogue record for this title is available from the British Library.

Get the most from this book

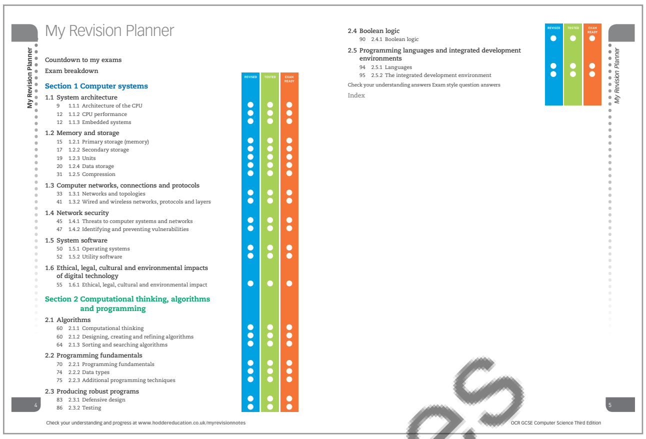
Everyone has to decide his or her own revision strategy, but it is essential to learn your work, review it and test your understanding. These Revision Notes will help you to do that in a planned way, topic by topic. Use this book as the cornerstone of your revision and don't hesitate to write in it – personalise your notes and check your progress by ticking off each section as you revise.

Tick to track your progress

Use the revision planner on pages 4 and 5 to plan your revision, topic by topic. Tick each box when you have:

- revised and understood a topic
- tested yourself
- practised the exam questions and gone online to check your answers.

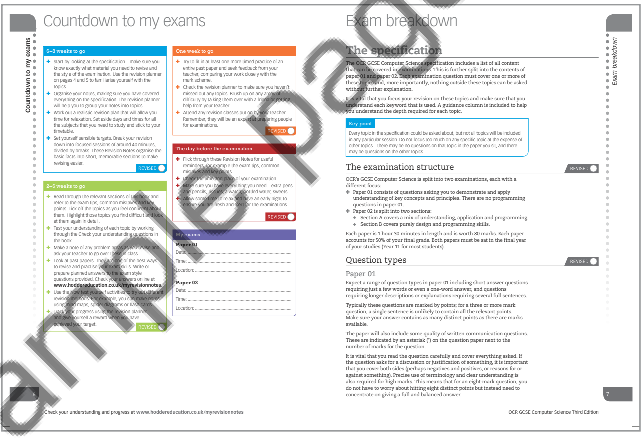
You can also keep track of your revision by ticking off each topic heading in the book. You may find it helpful to add your own notes as you work through each topic.



My Revision Planner

Check your understanding and progress at www.hoddereducation.co.uk/myrevisionnotes

OCR GCSE Computer Science Third Edition



Countdown to my exams

6–8 weeks to go

- Start by looking at the specification – make sure you know exactly what material you need to revise and the style of the examination. Use the revision planner on pages 4 and 5 to familiarise yourself with the topics.
- Organise your notes, making sure you have covered everything on the specification. The revision planner will help you to group your notes into topics.
- Work out a realistic revision plan that will allow you time for relaxation. Set aside days and times for all the subjects that you need to study and stick to your timetable.
- Set yourself sensible targets. Break your revision down into focused sessions of around 40 minutes, divided by breaks. These Revision Notes organise the basic facts into short, memorable sections to make revising easier.

REVISED ○

2–6 weeks to go

- Read through the relevant sections of this book and refer to the exam tips, common mistakes and key points. Tick off the topics as you feel confident about them. Highlight those topics you find difficult and look at them again in detail.
- Test your understanding of each topic by working through the Check your understanding questions in the book.
- Make a note of any problem areas as you revise and ask your teacher to go over these in class.
- Look at past papers. They are one of the best ways to revise and practise your exam skills. Write or prepare planned answers to the exam style questions provided. Check your answers online at **www.hoddereducation.co.uk/myrevisionnotes**.
- Use the Now test yourself activities to try out different revision methods. For example, you can make notes using mind maps, spider diagrams or flash cards.
- Track your progress using the revision planner and give yourself a reward when you have achieved your target.

REVISED ○

One week to go

- Try to fit in at least one more timed practice of an entire past paper and seek feedback from your teacher, comparing your work closely with the mark scheme.
- Check the revision planner to make sure you haven't missed out any topics. Brush up on any areas of difficulty by talking them over with a friend or getting help from your teacher.
- Attend any revision classes put on by your teacher. Remember, they will be an expert at preparing people for examinations.

REVISED ○

The day before the examination

- Flick through these Revision Notes for useful reminders, for example the exam tips, common mistakes and key points.
- Check the time and place of your examination.
- Make sure you have everything you need – extra pens and pencils, tissues, a watch, bottled water, sweets.
- Allow some time to relax and have an early night to ensure you are fresh and alert for the examinations.

REVISED ○

My exams

Paper 01
Date: ....
Time: ....
Location: ....

Paper 02
Date: ....
Time: ....
Location: ....

Exam breakdown

The specification

The OCR GCSE Computer Science specification includes a list of all content that can be covered in examinations. This is further split into the contents of paper 01 and paper 02. Each examination question must cover one or more of these topics and, more importantly, nothing outside these topics can be asked without further explanation.

It is vital that you focus your revision on these topics and make sure that you understand each keyword that is used. A guidance column is included to help you understand the depth required for each topic.

Key point

Every topic in the specification could be asked about, but not all topics will be included in any particular session. Do not focus too much on any specific topic at the expense of other topics – there may be no questions on that topic in the paper you sit, and there may be questions on the other topics.

The examination structure REVISED ○

OCR's GCSE Computer Science is split into two examinations, each with a different focus:

- Paper 01 consists of questions asking you to demonstrate and apply understanding of key concepts and principles. There are no programming questions in paper 01.
- Paper 02 is split into two sections:
  - Section A covers a mix of understanding, application and programming.
  - Section B covers purely design and programming skills.

Each paper is 1 hour 30 minutes in length and is worth 80 marks. Each paper accounts for 50% of your final grade. Both papers must be sat in the final year of your studies (Year 11 for most students).

Question types REVISED ○

Paper 01

Expect a range of question types in paper 01 including short answer questions requiring just a few words or even a one-word answer, and questions requiring longer descriptions or explanations requiring several full sentences.

Typically these questions are marked by points; for a three or more mark question, a single sentence is unlikely to contain all the relevant points. Make sure your answer contains as many distinct points as there are marks available.

The paper will also include some quality of written communication questions. These are indicated by an asterisk (*) on the question paper next to the number of marks for the question.

It is vital that you read the question carefully and cover everything asked. If the question asks for a discussion or justification of something, it is important that you cover both sides (perhaps negatives and positives, or reasons for or against something). Precise use of terminology and clear understanding is also required for high marks. This means that for an eight-mark question, you do not have to worry about hitting eight distinct points but instead need to concentrate on giving a full and balanced answer.

Check your understanding and progress at www.hoddereducation.co.uk/myrevisionnotes

OCR GCSE Computer Science Third Edition

Features to help you succeed

Exam tip

Expert tips to help polish your exam technique and maximise your chances in the exam

Common mistake

Common mistakes that candidates make and how to avoid them.

Check your understanding

Questions to test your understanding of basic facts.

Worked example

Worked examples illustrate methods, calculations and explanations.

Now test yourself

Activities to encourage note taking and revision.

Key point

Further explanation of some important issues.

Exam-style questions

Practice exam questions to consolidate your revision and practise your exam skills.

Definitions of **key terms** that need additional explanation are provided where they first appear.

My Revision Planner

REVISED TESTED EXAM READY

2.4 Boolean logic

2.5 Programming languages and integrated development environments

Countdown to my exams

6–8 weeks to go

- Start by looking at the specification – make sure you know exactly what material you need to revise and the style of the examination. Use the revision planner on pages 4 and 5 to familiarise yourself with the topics.
- Organise your notes, making sure you have covered everything on the specification. The revision planner will help you to group your notes into topics.
- Work out a realistic revision plan that will allow you time for relaxation. Set aside days and times for all the subjects that you need to study and stick to your timetable.
- Set yourself sensible targets. Break your revision down into focused sessions of around 40 minutes, divided by breaks. These Revision Notes organise the basic facts into short, memorable sections to make revising easier.

REVISED ○

2–6 weeks to go

- Read through the relevant sections of this book and refer to the exam tips, common mistakes and key points. Tick off the topics as you feel confident about them. Highlight those topics you find difficult and look at them again in detail.
- Test your understanding of each topic by working through the Check your understanding questions in the book.
- Make a note of any problem areas as you revise and ask your teacher to go over these in class.
- Look at past papers. They are one of the best ways to revise and practise your exam skills. Write or prepare planned answers to the exam-style questions provided. Check your answers.
- Use the Now test yourself activities to try out different revision methods. For example, you can make notes using mind maps, spider diagrams or flash cards.
- Track your progress using the revision planner and give yourself a reward when you have achieved your target.

REVISED ○

One week to go

- Try to fit in at least one more timed practice of an entire past paper and seek feedback from your teacher, comparing your work closely with the mark scheme.
- Check the revision planner to make sure you haven't missed out any topics. Brush up on any areas of difficulty by talking them over with a friend or getting help from your teacher.
- Attend any revision classes put on by your teacher. Remember, they will be an expert at preparing people for examinations.

REVISED ○

The day before the examination

- Browse through these Revision Notes for useful reminders, for example the exam tips, common mistakes and key points.
- Check the time and place of your examination.
- Make sure you have everything you need – extra pens and pencils, tissues, a watch, bottled water, sweets.
- Allow some time to relax and have an early night to ensure you are fresh and alert for the examinations.

REVISED ○

My exams

Paper 01

Date: ....

Time: ....

Location: ....

Paper 02

Date: ....

Time: ....

Location: ....

1.1 System architecture

A computer system consists of hardware and software working together to process data.

Hardware is the name for the physical components that make up the computer system.

Software is the name for the programs that provide instructions for the computer, telling it what to do.

A computer system receives information as an input, processes and stores that information, and then outputs the results of that processing.

The CPU processes the data.

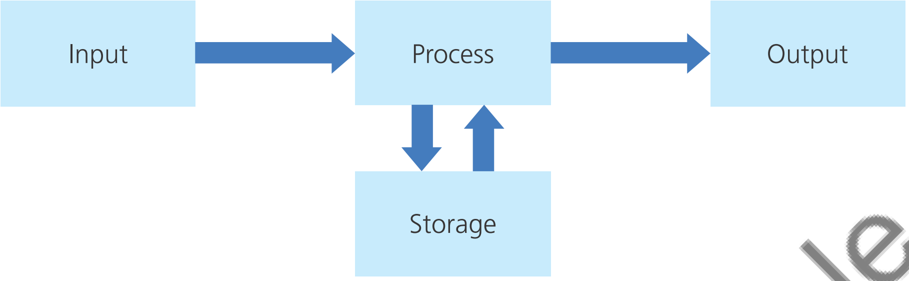


Figure 1.1.1 Input–process–output

1.1.1 Architecture of the CPU

Purpose of the CPU

REVISED ○

The purpose of the **CPU** is to carry out a set of instructions that is contained in a computer program.

It does this using the **fetch–execute cycle**.

- Fetch – an instruction in the form of data is retrieved from main memory.
- Decode – the CPU decodes the instruction.
- Execute – the CPU performs an action according to the instruction.

The CPU operates at high speeds governed by the clock chip.

- The clock chip uses a vibrating crystal to maintain a constant speed.
- The speed of the clock chip is measured in hertz, Hz (cycles per second) and typically works at up to 4 GHz (four billion cycles per second).
- The clock speed is the number of fetch–execute cycles per second.

The **CPU** is a collection of billions of electronic switches that process data, execute instructions and control the operation of the computer.

The **fetch–execute cycle** is the basic operation of the CPU. It continually fetches, decodes and executes instructions stored in memory.

Common CPU components and their function

REVISED ○

Arithmetic Logic Unit (ALU)

The ALU carries out the calculations and logical decisions required by the program instructions that the CPU is processing.

- Arithmetic operations, such as add and subtract.
- Logical operations, such as AND, OR and NOT, and the result of ‘less than’, ‘greater than’, ‘equal to’ comparisons.
- Binary shift operations, which are used for multiplication or division.

Control unit (CU)

The CU co-ordinates the activity of the CPU and memory in order to execute instructions. It:

- sends out signals to control how data moves around the CPU and memory
- decodes instructions from memory.

Cache memory

The purpose of cache memory is to provide temporary storage that the CPU can access very quickly.

- It stores instructions and data that are used repeatedly or are likely to be required for the next CPU operation.

Cache memory sits between the processor and main memory (RAM).

- The CPU looks in the cache for required data.
- If it is not there, it requests it from RAM.
- The data is moved into cache before being accessed by the CPU.

Registers

In **Von Neumann architecture**, data and instructions are stored in the same memory.

Von Neumann architecture is the most common organisation of computer components, where instructions and data are stored in the same place.

Typical Von Neumann architecture uses a number of registers.

Registers are memory locations within the CPU that hold data temporarily and can be accessed very quickly.

Their role in the CPU is to accept, store and transfer data and instructions for immediate use by the CPU.

Four of the registers found in the CPU are the ACC, PC, MDR and MAR.

Accumulator (ACC)

- Stores the results of any calculations made by the Arithmetic Logic Unit (ALU).
- Stores the value of inputs and outputs to and from the CPU.

Exam tip

Questions on these topics often require you to know these definitions.

Program counter (PC)

- Keeps track of the memory location (known as an address) for the next instruction.
- The program counter is incremented (increased by 1) to the next memory location at the fetch stage of the fetch–execute cycle, to allow the program to be executed line by line.
- Program instructions can modify the value in the program counter to alter the flow of the program so that it continues from a new location.

Memory data register (MDR)

- Stores any data fetched from memory or any data that is to be transferred to and stored in memory.

Memory address register (MAR)

- Stores the location in memory (an address) to be used by the MDR – that is, where the MDR needs to fetch data from or send data to.

Exam tip

You will often be asked about what type of information is held in each of these registers, an address or data or an instruction. Just saying data or address is not enough, you must explain more about the data or address.

Note: You do not need to know about buses for your examination.

Figure 1.1.2 is a simplified diagram showing the layout of these components, and how the CPU communicates with memory and input/output devices.

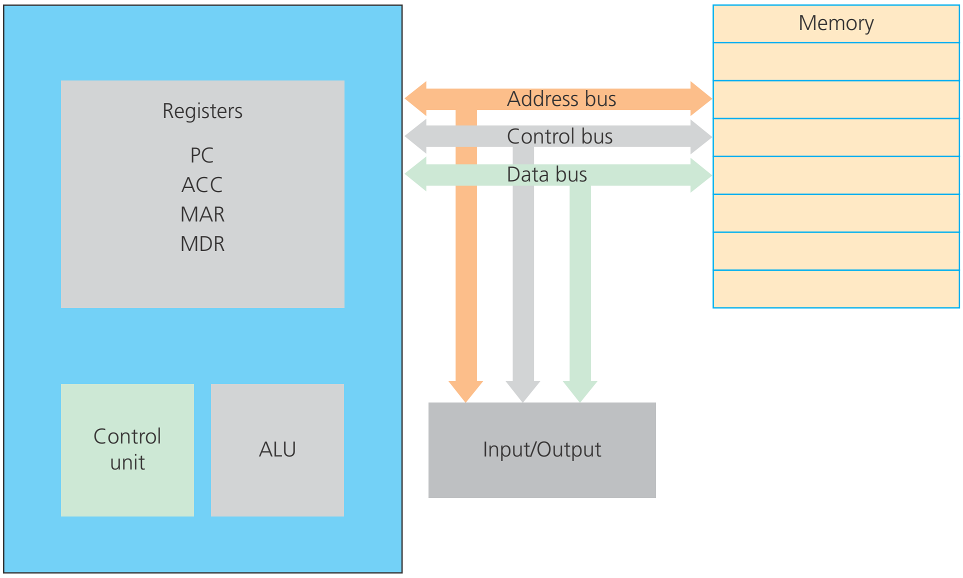


Figure 1.1.2 A CPU with Von Neumann architecture; the arrows represent the flow of data between components

The fetch–execute cycle in more detail

Fetch

1. Each instruction in a computer program is stored in a particular location (or address) in memory. The address of the next instruction is copied from the program counter and placed in the MAR.
2. The MAR now contains a memory address. The control unit fetches the data that is stored at that address and copies it to the MDR.
3. The program counter is incremented to point to the next instruction to be processed in the program, ready for the next fetch–execute cycle.

Decode

4. The MDR now contains either data to be processed by the CPU, fetched from memory, or an instruction. The control unit decodes the instruction to see what to do.

Execute

5. The decoded instruction is executed. This might mean performing a calculation using the ALU, locating some data in memory, changing the program counter value – or something else.

Once the execute part of the cycle is complete, the next fetch–execute cycle begins.

Exam tip

Make sure you know the three main stages of the fetch–execute cycle and that the PC is incremented at the fetch stage.

Revision activity

Load a web version of Little Man Computer (https://peterhigginson.co.uk/lmc/) and, from the box marked 'select', choose 'add'. This will load a simple program. Run this by selecting 'step' to see how the fetch–execute process uses the key registers in the CPU to add together two numbers input by the user.

Check your understanding

1. What is the purpose of the CPU?
2. What are the **three** main stages in the fetch–execute process?
3. Identify **three** registers and what type of data they hold during the fetch–execute cycle.
4. State the key feature of Von Neumann architecture.

Answers on p. 98

1.1.2 CPU performance

How common characteristics of CPUs affect their performance

REVISED ○

Clock speed

The CPU is constantly fetching and executing instructions.

The speed at which it does this is determined by an electronic clock.

The faster the clock speed, the more instructions that can be executed every second.

The clock speed is measured in Hertz; 1 Hz = once per second.

Exam tip

You need to know how all of these factors work together to determine how quickly the computer works.

Amount of cache memory

Cache memory is located between the main memory and the CPU.

It is used to hold data that needs to be accessed very quickly.

Accessing cache memory is much faster than accessing main memory.

The larger the cache memory, the more likely it is that the required data will have been copied from main memory.

The more cache memory available, the better the performance of the computer.

Key point

Cache memory is significantly more expensive than main memory (RAM) and a typical computer will only have KB of cache compared to GB of RAM.

Number of processor cores

In a multi-core CPU there are a number of processor cores each capable of carrying out the fetch–execute cycle.

If the program supports multi-cores, the CPU can handle several instructions at the same time.

Check your understanding

5 Describe the **three** factors that affect the performance of the computer.

Answers on p. 98

Exam tip

Having multiple cores does not necessarily improve the performance of the computer. This only works if the program has been developed to use multiple cores and the program running is capable of being split into subsections.

1.1.3 Embedded systems

The purpose and characteristics of embedded systems

REVISED ○

An **embedded system** is a computer system that has a dedicated function as part of a larger device.

When a computer device is required to perform a single or fixed range of tasks, it can be engineered to reduce its size and complexity in order to focus only on these tasks.

Dedicated software will be programmed into the device to complete the necessary tasks and nothing else.

The reduction of complexity of the hardware and the dedicated nature of the software will make the device more reliable and cost-effective than using a general-purpose computer.

The main components of a computer are either manufactured onto a single chip (a microcontroller), or separate circuits for processing and memory are combined into a larger device.

The embedded system will typically include some ROM (read-only memory) to store the dedicated program and some RAM to store user inputs and processor outputs.

Embedded systems have the following characteristics.

- **Low power** so they can operate effectively from a small power source such as in a mobile phone.
- **Small size** so they can fit into portable devices such as a personal fitness device.
- **Rugged** so that they can operate in harsh environments such as car engine management systems or in military applications.
- **Low cost** so that they are suitable for use in mass-produced, low-cost devices such as microwave ovens.
- **Dedicated software** to complete a single task or limited range of tasks, such as in computer aided manufacture or control systems.

Examples of embedded systems

Embedded systems are found within common household devices such as:

- washing machines
- set-top boxes
- telephones
- televisions
- home security and control systems.

Embedded systems are also widely used within larger and more complex systems, such as:

- car engine management
- airplane avionics
- computer-controlled manufacturing
- military applications such as guidance systems.

Embedded systems are frequently connected to the internet via Wi-Fi to exchange data with third parties or apps on other devices, such as:

- water meters
- energy smart meters
- home security
- central heating management systems.

Examples of embedded system inputs and outputs:

System	Input examples	Output examples
washing machine	choice of program	display progress
	water temperature	signal to heater
	water level	signal to water input valve
satnav	destination	driving instructions
	GPS position	current location mapped

Exam tip

It is useful to think about what data is input and output by common devices with embedded systems and what is held in ROM and RAM. Think also about the environment in which the device operates to identify which features of the embedded system make them appropriate.

Check your understanding

6 Describe **three** features that make an embedded system appropriate for use in a small drone helicopter.

Answers on p. 98

Exam checklist

In this chapter you learned about:

The purpose of the CPU

- To carry out a set of instructions contained in a computer program using the fetch–execute cycle

Common CPU components and their function

- ALU to carry out arithmetic calculations and logical decisions
- CU to decode instructions and control how data moves in the CPU to execute the instructions
- Cache memory to temporarily hold instructions and data that the CPU is likely to need
- Registers, which are memory locations within the CPU that hold data

Von Neumann architecture

- In Von Neumann architecture, data and instructions are stored in the same memory
- Four key registers in Von Neumann architecture are:
  - Accumulator to store the results of calculations carried out by the ALU
  - Program counter to hold the address of the next instruction
  - Memory data register to hold the data fetched from memory or to be sent to memory
  - Memory address register to hold the address of the next memory location to be accessed

How common characteristics of CPUs affect their performance

- Clock speed
- Cache memory
- Number of cores

The purpose and characteristics of embedded systems

- Designed for a dedicated function as part of a bigger system
- Often manufactured as a single chip
- Dedicated hardware and software to perform a limited set of tasks
- Programs often uploaded at manufacturing stage
- Limited options to modify the programs
- Low power consumption
- Small
- Rugged
- Low cost

Now test yourself

TESTED ○

1 Make a list of the main components of the CPU and what they do.

2 List the stages of the fetch–execute cycle and what happens at each stage.

3 Make a list of the factors that affect the speed of the CPU.

4 List some embedded systems and identify the inputs and outputs.

Exam-style questions

1 A computer is advertised as having a clock speed of 2.8 GHz, 2.5 MB cache and four cores.

a) Describe how the clock speed affects the performance of the computer. [2]

b) Describe why having more cache memory will improve the performance of the computer. [2]

c) State the purpose of the memory address register (MAR). [2]

d) Explain how a multi-core CPU can improve the performance of the computer. [2]

2 a) State **one** item that might be held in the ROM in an embedded system inside a washing machine. [1]

b) State **two** items of data that might be held in RAM in an embedded system inside a washing machine. [2]

c) Describe **two** important features of an embedded system that make it appropriate for use in a car engine management system. [4]

3 a) Describe **three** types of operation carried out by the ALU. Give an example for each one. [6]

b) Describe what happens at the fetch stage of the fetch–execute cycle. [3]

Answers on p. 103

INDEX

MY **REVISION** NOTES

OCR GCSE (9–1)

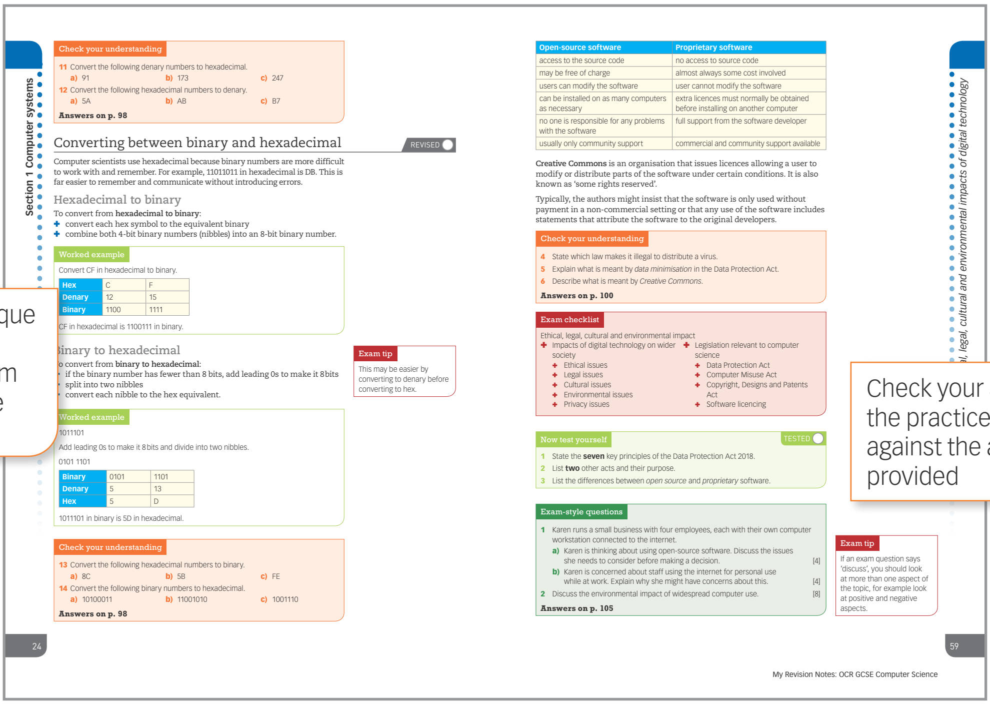
COMPUTER SCIENCE

Target exam success with *My Revision Notes*. Our updated approach to revision will help you learn, practise and apply your skills and understanding. Coverage of key content is combined with practical study tips and effective revision strategies to create a guide you can rely on to build both knowledge and confidence.

***My Revision Notes: OCR GCSE (9–1) Computer Science* will help you:**

Strengthen subject knowledge and key terms by working through clear and focused key content

Test understanding and identify areas for improvement with 'check your understanding' questions



Section 1 Computer systems

Check your understanding

11 Convert the following denary numbers to hexadecimal.
a) 91 b) 173 c) 247

12 Convert the following hexadecimal numbers to denary.
a) 5A b) AB c) B7

Answers on p. 98

Converting between binary and hexadecimal

REVISED ○

Computer scientists use hexadecimal because binary numbers are more difficult to work with and remember. For example, 11011011 in hexadecimal is DB. This is far easier to remember and communicate without introducing errors.

Hexadecimal to binary

To convert from **hexadecimal to binary**:

- convert each hex symbol to the equivalent binary
- combine both 4-bit binary numbers (nibbles) into an 8-bit binary number.

Worked example

Convert CF in hexadecimal to binary.

Hex	C	F
Denary	12	15
Binary	1100	1111

CF in hexadecimal is 1100111 in binary.

inary to hexadecimal

o convert from **binary to hexadecimal**:

- if the binary number has fewer than 8 bits, add leading 0s to make it 8 bits
- split into two nibbles
- convert each nibble to the hex equivalent.

Exam tip

This may be easier by converting to denary before converting to hex.

Worked example

1011101

Add leading 0s to make it 8 bits and divide into two nibbles.

0101 1101

Binary	0101	1101
Denary	5	13
Hex	5	D

1011101 in binary is 5D in hexadecimal.

Check your understanding

13 Convert the following hexadecimal numbers to binary.
a) 8C b) 5B c) FE

14 Convert the following binary numbers to hexadecimal.
a) 10100011 b) 11001010 c) 1001110

Answers on p. 98

24

Open-source software	Proprietary software
access to the source code	no access to source code
may be free of charge	almost always some cost involved
users can modify the software	user cannot modify the software
can be installed on as many computers as necessary	extra licences must normally be obtained before installing on another computer
no one is responsible for any problems with the software	full support from the software developer
usually only community support	commercial and community support available

Creative Commons is an organisation that issues licences allowing a user to modify or distribute parts of the software under certain conditions. It is also known as 'some rights reserved'.

Typically, the authors might insist that the software is only used without payment in a non-commercial setting or that any use of the software includes statements that attribute the software to the original developers.

Check your understanding

4 State which law makes it illegal to distribute a virus.

5 Explain what is meant by *data minimisation* in the Data Protection Act.

6 Describe what is meant by *Creative Commons*.

Answers on p. 100

Exam checklist

Ethical, legal, cultural and environmental impact

- Impacts of digital technology on wider society
  - Ethical issues
  - Legal issues
  - Cultural issues
  - Environmental issues
  - Privacy issues
- Legislation relevant to computer science
  - Data Protection Act
  - Computer Misuse Act
  - Copyright, Designs and Patents Act
  - Software licencing

Now test yourself

TESTED ○

1 State the **seven** key principles of the Data Protection Act 2018.

2 List **two** other acts and their purpose.

3 List the differences between *open source* and *proprietary* software.

Exam-style questions

1 Karen runs a small business with four employees, each with their own computer workstation connected to the internet.

a) Karen is thinking about using open-source software. Discuss the issues she needs to consider before making a decision. [4]

b) Karen is concerned about staff using the internet for personal use while at work. Explain why she might have concerns about this. [4]

2 Discuss the environmental impact of widespread computer use. [8]

Answers on p. 105

Exam tip

If an exam question says 'discuss', you should look at more than one aspect of the topic, for example look at positive and negative aspects.

al, legal, cultural and environmental impacts of digital technology

59

My Revision Notes: OCR GCSE Computer Science

Enhance exam technique through exam-style questions and tips from leading author George Rouse

Check your answers to the practice questions against the answers provided

Plan and manage a successful revision programme with the 'exam breakdown', 'countdown to the exams' and 'now test yourself' sections

HODDER EDUCATION
t: 01235 827827
e: education@hachette.co.uk
w: hoddereducation.co.uk

ISBN 978-1-3983-2114-4

9 781398 321144

FSC