2019-2021

Woodbridge High School Computer science OCR A-level Booklet

Name: TG:

Table of Contents

Welcome

Welcoming to the Computing department at Woodbridge High School, this is the introduction work booklet for students starting their A-level for Computer Science

The Computing department is made up of 4 computer science teachers.

Mr Brock Carey - Room R15

Mr Arday - Room R7

Mr Khan - Room R14

Lessons at A-level

For the A-level course for the 2018-2020 cohorts, you will be taught by two teachers.

You will have nine lessons a fortnight in total, 6 with Mr Brock-Carey and 3 with Mr Khan. These will be based in the Roding Arts building in room r14 and r15.

Teacher Contacts

dbrock-carey@woodbridgehigh.co.uk

wkhan@woodbridgehigh.co.uk

Course Content

Learners must take three components (01, 02 and 03) to be awarded the OCR A Level in Computer Science.

Computer systems (01) 140 marks 2 hours and 30 minutes written paper (no calculators allowed) 40%

Algorithms and programming (02) 140 marks 2 hours and 30 minutes written paper (no calculators allowed) 40%

Programming project (03) 70 marks Non-exam assessment 20%

What do I need for lessons?

You will be given two assessment folder where all assessed work one for each teacher. This will mostly be written work (tests and any work that you have been given feedback on) but will also include your tracker, where your teacher will ask you to record the feedback they give you. This file will stay in the Computing Department.

You will need to provide your ring binder file for your class notes. You should have this with you in every lesson as you may need to add to or refer to previous notes.

What exams will I have?

Two exams, externally assessed

Paper 1 Computer Systems

- 2-hour 30-minute exam
- 140 marks
- Short and long answer
- Candidates may be required to write Algorithms

Paper 2 Algorithms and programming

- 2-hour 30-minute exam
- 140 marks
- Section A
    - Short and long answer questions
- Section B
    - Questions based around one scenario
- Candidates will need to write pseudocode/program code

What am I expected to do outside of lessons?

Outside of the lesson, you will need to be programming and reading more. You should be spending 9 hours a week across both programming and theory practice.

The programming part will need not to be done weekly to improve your skills for your projects.

There will also be home learning frequently set by your teachers. This will often be essay practice questions or task sheets. Sometimes you may be asked to complete a research task or prepare a presentation.

What about Extra-Curricular?

As an A Level Computing student, you have an important role to play in the Extra-curricular life of the Computing Department. We will be running extra classes for the student taking GCSE and Coding clubs for the younger years of the school.

In return, you will be able to use the computing room when they are available, and this will also improve your skills that are important for the project that you will be making at the end of year 12.

What can I do to get prepared for September?

For September you will need to learn python programming. The reason for this if you cannot show good progress and skill with this language, then you will need to do this on top of your regular workload in September before you move on to programming in C# etc.

It would be best if you also went over the GCSE course and the work included in this booklet. This will help you with your programming and theory revision to help you get the best start for your A-levels.

Specification Information

A Level GCE Computer Science - H446

OCR A-Level Computer Science qualification helps students understand the core academic principles of computer science. Classroom learning is transferred to creating real-world systems through the creation of an independent programming project. OCR A-Level will develop the student's technical understanding and their ability to analyse and solve problems using computational thinking

Departmental Sanctions

In the computing department, we will be using the schools Sixth Form Intervention Policy. This means that if you need help, you will be given it, but it can lead to measures being put in place if you are not working an following the rules of the school.

Content Overview

A Level	
Component 1: Computer Systems	**Component 02: Algorithms and Problem Solving**
Structure and Function of Processor	Thinking Abstractly
Types of Processor	Thinking Ahead
Input, Output and storage	Thinking Procedurally
Systems Software	Thinking Logically
Software Development	Thinking Concurrently
Types of Programming Language	Programming Techniques
Compression, Encryption and Hashing	Computation Methods
Databases	Algorithms
Networks	
Web Technologies	**Component 03: Programming Project**
Data Types	Analysis of the problem (10 marks)
Data Structures	Design of the solution (15 marks)
Boolean Algebra	Developing the solution (25 marks)
Computing Related Legislation	Evaluation (20 marks)
Ethic, moral and cultural issues	

Google classroom

All work will be put on to google classroom the class code: efvb4t. You will upload your work when it is complete.

Programming task

This task is set and to be completed like the GCSE NEA as this is how the A-level project is set and marked.

The Redbridge Parking Company would like a program that can be installed into its parking meter. The meter has a keypad for typing in details and an OK and cancel button. The meter is linked to a file (database) that stores all of the aspect of the information that a user enters as well as the time and payments details.

The system that you will make is as the following:

- Display a message to welcome users to the car park and what the hourly rate is for parking.
- Allow the user to input the last three characters of their license plate.
- Ask the user for how long they want to park for.
- Allow the parking meter to say how much is needed to be paid, e.g. £10.00.
- Allow the user to enter how much they have paid
- Allow the meter to work out and state how much is left to pay.
- Allow the meter to indicate if any change is given.
- Print out on screen a parking ticket for a car parking space.
- Print the last three characters on to the parking ticket.
- The system should save the car, payment details and parking times to a TXT or CSV file

The write up should have the following parts.

- **Analysis (6 marks)** - breaking down the task down, working out what it needs to do
- **Design (8 marks)** - planning how you are going to solve the problem and test it, this includes algorithms
- **Development (8 marks)** - showing how you solved the problem and how you developed it bit by bit, also showing how you tested it as you made it.
- **Testing, Evaluation & Conclusions (6 marks)** - testing your finished solution thoroughly to show it work, writing a conclusion against the success criteria.

Analysis section

This should include:

1. An explanation of the task that you need to complete, saying what it will need to do
2. A breakdown of the main task, listing the smaller sub-tasks

Design section

This should include:

1. Success criteria - a list of success criteria for the requirement.
2. Variables to be used - this should include the data type and what each one will be used for.
3. Validation to be used - this should explain the different validation that could/will be added to the solution.
4. Pseudocode & flowchart of the solution to be created.

5. Test plan - this should detail the tests that will be carried out after development.

Development section

This should include:

1. Explanation showing how you developed the code bit by bit to get to your finished solution.
2. Full annotation of the final code with screenshots of it.

Testing section

1. The test plan has been fully completed with screenshots showing the tests being carried out.
2. If any tests failed, these should be corrected and re-tested.

Evaluation

1. You must evaluate your final program against the success criteria you came up with at the start.

Computer Science Theory

Recommended resources

- GCSE textbooks, as recommended by your school
- teach-ict.com (may require a school login)
- PiXL Computer Science 8 – 9 Revision Pack
- PiXL Computer Science Booster Pack

Wider computing issues and integrated questions

These questions require you to use your technical knowledge in context. Reference any sources that you use to help you.

1. Create a timeline showing the history of computing, including any key discoveries or inventions. Extend your timeline to show how you think computer science might develop over the next 50 years.

2. Compare the Xbox ONE, PS4 Pro and PC as gaming platforms. You must use as much technical detail as possible and reference any evidence presented. Choose how you will present your ideas.

3. Discuss the benefits and limitations of Virtual Reality
    a. In business contexts, such as medicine
    b. As a gaming tool
    c. As an extension to social media

4. Design the next piece of wearable technology, annotating how it will function and explain the function and purpose of any components used.

Systems Architecture

1. Produce an annotated diagram showing how the CPU processes data. This should include
    a. The purpose of the CPU
    b. Common CPU components and their function
        i. Arithmetic and Logic Unit (ALU)
        ii. Control Unit (CU)
        iii. Cache
        iv. Registers
            1. Memory Address Register (MAR)
            2. Memory Data Register (MDR)
            3. Program Counter
            4. Accumulator
    c. Reference to the fetch-execute cycle
2. Discuss, with examples, how the performance of a CPU can be improved, including:
    a. Increasing the clock speed
    b. Increasing the cache size
    c. Increasing the number of processing cores

Memory

1. Compare RAM and ROM

2. Explain the need for virtual memory in a computer system

3. Describe the characteristics of flash memory

Storage

1. Complete the following table comparing optical, magnetic and solid-state storage media

	Capacity	Speed	Portability	Durability	Reliability	Cost
Optical						
Magnetic						
Solid State						

2. Justify one use of each storage method

Networks

1. Explain the similarities and differences between
    a. A LAN and a WAN
    b. Client-server and peer-to-peer networks
2. Explain the difference between the Internet and the World Wide Web
3. Describe the factors that affect network performance, and explain how network performance can be improved
4. Draw three different network topologies
    a. Label all the components required to create each network
    b. Explain the purpose of each component in the network, including
        i. Wireless Access Points
        ii. Routers
        iii. Switches
        iv. Network Interface Cards
        v. Transmission media, such as Ethernet Cables
5. Create an interactive presentation or resource, such as a website, that explains how your computer connects to a webpage, such as www.bbc.co.uk Include the following:
    a. DNS
    b. Hosting
    c. TCP/IP, including the concept of layers
    d. HTTP/HTTPS
    e. Packet switching
6. There have been many recent high-profile cyber-attacks across the world, including the attack on the NHS in May 2017. Some commentators have said that "we now rely too much on technology". Write an essay explaining how far you agree with this statement and including descriptions of threats to IT systems and ways to reduce vulnerabilities.

Systems Software

1. Create a presentation comparing Windows, Linux, iOS, Android (which is based on Linux) and Unix. Discuss the features of each operating system, comparing the benefits and limitations of each. Note that you can try a basic Unix interface here: http://www.masswerk.at/jsuix/

Ethical, Legal, Cultural and Environmental Concerns

Find a recent news story on one of the following topics:

- A legal issue in computing, such as a breach of the Data Protection Act
- An ethical issue in computing, such as the development of AI
- An environmental issue in computing, such as the disposal of waste equipment
- A technical development in computer science, such as the Internet of Things

Summarise the story, explaining any technical content for a student in year 10.
Explain how the story affects you as a student of computer science.

Computational Thinking – Theory

Computational Logic and Calculations

1. Complete the truth tables for the following expressions
    a. A AND (B OR C)

A	B	C	B OR C	A AND (B OR C)
0	0			
0	0			
0	1			
0	1			
1	0			
1	0			
1	1			
1	1			

b. (NOT A) OR (NOT B)
    i. What single logic gate produces the same result as this expression?

A	B			

c. Draw a circuit to represent each expression

2. Calculate each of the following, showing any appropriate working you need
    a. 13 MOD 2
    b. 16 MOD 6
    c. 15 MOD 3
    d. 7 MOD 8
    e. 13 DIV 2
    f. 16 DIV 6
    g. 15 DIV 3
    h. 7 DIV 8
    i. 2^0
    j. 2^7
    k. 2^8
    l. 2^10

3. Covert the following into the units given
    a. 4 bytes = bits
    b. 1 TB = bytes
    c. 80 kB = GB
    d. 40 MB = nibbles

4. Complete the table, converting between binary, hexadecimal and denary as required

Binary	Hex	Denary
0010 1010		
	0B	
		255
0110 0111		
	F5	
		48
	CD	

5. Complete the following calculations
    a. 0110 0011 + 0011 0001
    b. 1010 0110 + 1100 1111
    c. 0110 0011 << 2 (bit shift left two places)

6. Check if these are valid ASCII characters. If they are, give their character equivalent. Note that the first bit is a check digit using even parity, and the remaining 7 bits are the character
    a. 1110 0010
    b. 1100 0111
    c. 0011 0110
    d. 1100 1010

Programming Tools and Standards

1. Compare the use of jpg, png and gif to store images, explaining the benefits, properties and uses of each image format

2. Produce an annotated diagram of the IDE you prefer to use to write code, explaining any features of the IDE that help you to produce your code. You may need to show examples of the IDE in use to highlight the different features.

GCSE RAG Theory

Rag your GCSE knowledge against the Specification to show you what you need to go over by September

Component 1		RAG
1.1 Systems Architecture	I can explain the purpose of the CPU	
	I can describe the components of Von Neumann Architecture	
	I can explain the role and operation of the main memory and the major components of the CPU (Control Unit, ALU, Bus, Cache)	
	I can explain the stages of the Fetch-Execute Cycle	
	I can explain the effect of clock speed, number of cores, cache size and type on the performance of the CPU	
	I can explain the purpose of an embedded system and give examples	
1.2 Memory	I can explain the difference in purpose between RAM and ROM	
	I can explain a computer's need for virtual memory	
	I can describe flash memory	
1.3 Storage	I can explain the need for secondary storage	
	I can calculate data capacity requirements	
	I can explain the operation of and advantages and disadvantages of the following storage devices: Optical, Magnetic, Solid Sate	
	I can analyse the advantages and disadvantages of the above storage devices for a given scenario/application	
1.4 Wired and Wireless Networks	I can describe the following types of networks: PAN, LAN and WAN	
	I can explain the factors that affect the performance of a network	
	I can explain the roles of computers in client-server network, peer-to-peer network	
	I can explain the concepts: Domain Name Server, Hosting, The Cloud	
	I can explain the concept of virtual networks	
1.5 Network Topologies, Protocols and Layers	I can describe a star and mesh network topology	
	I can explain how Wifi uses: frequency and channels	
	I can explain how Wifi uses: encryption	
	I can explain IP addressing	
	I can explain MAC addressing	
	I can explain the purpose and use of the following protocols:	
	Ethernet	
	Wifi	
	TCP/IP	
	UCP	
	HTTP/ HTTPS	
	FTP	
	POP	
	IMAP	
	SMTP	
	I can describe the 4 layers TCP/IP model	
	I can explain how packet switching is used to help transmit data	

Component 1		RAG
1.6 System Security	I can describe different forms of attack on a system/network	
	I can describe the main threats posed to networks	
	I can describe the main types of utility systems software	
	I can describe the roles and methods of the following types of backup: full, incremental	
1.7 Systems Software	I can explain the purpose and functionality of systems software	
	I can explain the purpose, features and functions of operating systems	
	I can describe the main types of utility systems software	
	I can describe the roles and methods of the following types of backup: full, incremental	
1.8 Ethical, Legal, Cultural and Environmental Concerns	I can explain the following legislations:	
	The Data Protection Act 1998	
	Computer Misuse Act of 1990	
	Copyright Designs and Patents Act 1988	
	Creative Commons Licensing	
	Freedom of Information Act 2000	
	I can compare Open source and Proprietary Software	
	Given a scenario, I can explain how key stakeholders are affected by technology	
	I can write an extended answer discussing the ethical, legal, cultural, environmental and privacy issues of a scenario or type of technology	

GCSE RAG Algorithms

Rag your GCSE knowledge against the Specification to show you what you need to go over by September

Component 2		RAG
2.2 Programming Techniques	I can explain the terms: Abstraction, decomposition and algorithmic thinking	
	I can compare and contrast binary and linear search algorithms	
	I can compare and contrast bubble, merge and insertion sort algorithms	
	I can produce an algorithm in Pseudocode or a Flow Chart that successfully solves a problem	
	I can interpret, correct or complete an algorithm	
2.2 Programming Techniques	I can identify when to use the following data types: Integer, Boolean, Real, Character and String	
	I can declare variables and constants with meaningful identifier names	
	I can use local and global variables appropriately	
	I can use selection when programming	
	I can create a subroutine (procedures and functions)	
	I can use parameters to pass data within a program	
	I can use definite and indefinite iteration	
	I can use nested selection and nested iteration	
	I can use arithmetic operations in a programming language	
	I can use relational operations in a programming language	
	I can use Boolean operations in a programming language	
	I can use Boolean operations in a programming language	
	I can create an array/list in a programming language (up to a 2d array)	
	I can create a record in a programming language	
	I can input data from a file to a program	
	I can output data from a program to a file (write and append)	
	I can use string handling operations in a programming language	
	I can use a random number generator in a programming language	
	I can use SQL to search for data	
2.3 Producing Robust Programs	I can include the following in my programming: Input sanitisation/validation, Plan for contingencies, anticipate misuse, authentication	
	I understand how comments and indentation improve the maintainability of a program	
	While using suitable test data I can apply the following types of testing: iterative, final/terminal	

Component 2		RAG
2.4 Computational Logic	I can explain why data is represented in computer systems in binary form	
	I can create/amend/complete simple logic circuits/statements using the operations AND, OR and NOT	
	I can describe the main types of utility systems software	
	I can complete a trace table to determine the purpose of an algorithm	
2.5 Translators and Facilities of Languages	I can explain the characteristics and purpose of high and low-level languages	
	I can explain the purpose of translators	
	I can describe the characteristics of: an assembler, a compiler and interpreter	
	I can describe the common tools of an IDE	
2.6 Data Representation	I know and can convert the units of information from bit to Petabyte	
	I understand and can convert between binary, hexadecimal and decimal	
	I can add two 8 bit binary integers and explain overflow errors that may occur	
	I can explain the term check digit	
	I can explain the advantages and disadvantages of using different character sets to represent data	
	I can explain how bitmap images are represented in binary by explaining the terms pixel, resolution and colour depth	
	I can calculate a bitmap image file size based on the number of pixels and colour depth	
	I can explain what metadata is	
	I can explain how the sample rate and sample resolution represent sound digitally	
	I can calculate the size of a sound file based on the sample rate and sample resolution	
	I can explain the need for data compression and methods of compressing data (lossless and lossy)	