

TRANSITION WORK



COMPUTER SCIENCE 2020

HoD: Richard Herbert

Email: rherbert@guildfordcounty.co.uk

Computer Science is a challenging A Level subject which includes a combination of theory and practical work. The best computer scientists enjoy thinking about and solving complex logical problems and puzzles: they are good at maths and physics and like to *think* computationally.

At A Level, we continue to use the programming constructs learned at GCSE: sequence; selection; and iteration. Writing code is an essential skill in the modern world, and at A Level, we build upon the foundations laid at GCSE and learn to program in Java, becoming more object-oriented in our approach.

The course covered is **AQA A Level Computer Science 7517**, and the link to specification can be found below.

Reading list (books / academic articles / journals etc)

Reading is essential. If you do not read, then you are already putting yourself at a massive disadvantage. Students who read regularly and read around the topics covered in class, are much more likely to achieve higher grades.

All these books can be found on Amazon, and occasionally on eBay.

Whilst I respect the environment, it is always recommended to have a hard copy of a textbook (as opposed to an electronic copy or PDF). This way you can always step away from your computer, get comfortable, and **READ**.

Essential Reading

AQA A level Computer Science, Bob Reeves

The Pattern on the Stone: The Simple Ideas That Make Computers Work,

Daniel Hillis

Desired Reading

AQA AS and A Level Computer Science, PM Heathcote

A Level Computer Science for AQA Unit 1, Kevin Bond

A Level Computer Science for AQA Unit 2, Kevin Bond

Above and Beyond Reading

Tackling A Level Projects in Computer Science AQA 7517, PG Online

Documentaries

On Netflix:

The Great Hack (documentary about the Cambridge Analytica scandal)

Inside Bill's Brain (documentary on Bill Gates)

The Social Network (drama on how Facebook was created)

Elsewhere:

Lo and Behold (documentary on how the Internet has changed the world)

Silicon Cowboys (documentary about Compaq vs IBM)

Brexit: the Uncivil War (drama about Cambridge Analytica)

Imitation Game (film drama about the great Alan Turing)

For the module *Consequences of Uses of Computing* you must have a good grasp of the ethical, legal, cultural, and environmental impact of technology (both positive and negative), and all the issues that surround it.

Watching documentaries and the news to keep up with the constant changes in technology, the influence technology has on our lives and society, as well as the laws that govern and control how technology is used responsibly is crucial.

Use the *BBC Technology* section as a good starting point for this:

<https://www.bbc.co.uk/news/technology>

Podcasts and Online Resources

Craig n Dave GCSE OCR Spec Order

<https://www.youtube.com/playlist?list=PLCiOXwirraUAvkTPDWeeSqAKty3LAG37->

Craig n Dave A Level AQA Spec Order

https://www.youtube.com/playlist?list=PLCiOXwirraUDUYF_qDYcZV8Hce8dsE_Ho

BBC Bitesize

<https://www.bbc.co.uk/bitesize/examspecs/zmtchbk>

Isaac Computer Science

<https://isaacomputerscience.org/>

Seneca Learning

<https://senecalearning.com/>

BBC Technology

<https://www.bbc.co.uk/news/technology>

GCSE OCR 276 Spec

<https://www.ocr.org.uk/qualifications/gcse/computer-science-j276-from-2016/>

A Level AQA 7517 Spec

<https://www.aqa.org.uk/subjects/computer-science-and-it/as-and-a-level/computer-science-7516-7517>

Written Work Required

In order to prepare to “*hit the ground running*” in September, you must be confident in all the topics already covered at GCSE, including the topics (from *OCR J276 Spec*) we were about to cover pre-Covid19 below:

- 1.1 Systems Architecture*
- 1.6 System Security*
- 1.8 Ethical, Legal, Cultural and Environmental Concerns*
- 2.1 Algorithms*
- 2.3 Producing Robust Programs*
- 2.5 Translators and Facilities of Language*

All the tasks below are to be completed before September, and the two completed documents emailed to rherbert@guildfordcounty.co.uk. *All four tasks will take you a minimum of 15 hours. **Remember**: the best students study consistently, over a long period of time, and put the effort in! *The sooner you start, the better.**

Task 1

Copy and paste your current *Class Notebook* notes (or from your other notes, see below *) into a Word document, so that they are all in one place, and readily accessible next year. Put the sections in the same order as they are on the specification (link above) and create an indexed contents page.

* If you are an external student (i.e. you were not in GCS Year 11), please copy all your notes from your previous school into a Word document. If you do not have access to these notes, you must make a summary for all the sections in the OCR J276 specification.

Task 2

Starting with *1.1 Systems Architecture* (as above), please spend at least two hours on each section, making detailed notes using the same Word document from *Task 1*. Use a combination of research from your GCSE CGP textbook, Craig n Dave videos (playlist above), BBC Bitesize, and any other online resources. If you have the CGP textbook, then for each section, answer the exam questions and check your answers. Copy your results into the Word document.

Task 3

Check your emails for an invitation to *Isaac Computer Science* and follow this link to create an account. Complete all the assignments that have been set to test your knowledge learned from *Task 2*.

Task 4

Complete the *Introduction to JAVA* (following pages).

Introduction to JAVA

To get a head start in *A Level Computer Science* it is important to learn Java programming as much as possible. By practicing **BEFORE** September you stand a **MUCH** greater chance of doing really well and achieving a higher grade. In fact, it is likely to help you so much that if you complete the tutorials and activities in this document it is estimated that you will achieve **1 to 2 grades higher** in your A Level!

This document will take you only one to three hours to complete.

Please email rherbert@guildfordcounty.co.uk or meleftheriadi@guildfordcounty.co.uk if you need more information or guidance.

Lesson and homework tasks throughout the course are differentiated with **Beginner**, **Intermediate** and **Expert** tasks, meaning that if you have completed all the basics before starting the course, you will be well prepared for going straight for the **Intermediate** and **Expert** tasks!

During A Level a tracking sheet is used to show your progress in both programming and the theory topics. A leader-board in the classroom is used to display those students who have made the most progress.

Do not worry if you get stuck on a task or cannot get your code to work: just paste your code into the appropriate section below and we will discuss it in September. Remember, a computer is not intelligent, and can only do what you, the programmer, tell it to do – they need very precise and specific instructions!

Failing whilst programming, and using a trial and error approach, is the best way to learn! Success comes after failing, sometimes multiple times.

It is the people who are willing to get things wrong and keep going who find the most success in life!

There are many excellent tutorials online to help get you prepped and ready to do brilliantly.

At A Level we use the **Java** programming language and use **Net Beans** integrated development environment to help us write our **Java** programs.

To follow the links in this document, hold down the Ctrl key when you click on them or right click and do "Open Hyperlink".

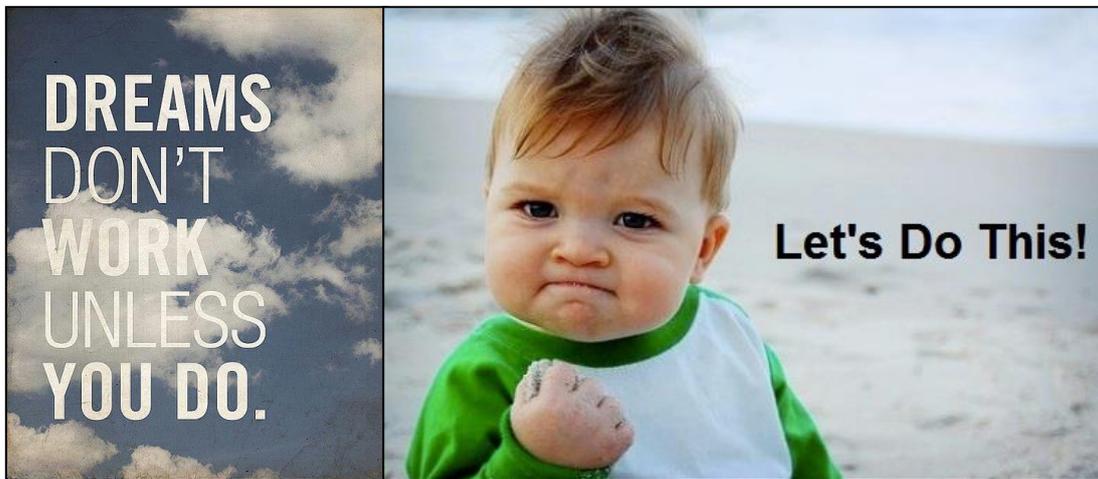
Work your way through the tutorial videos – pasting evidence of your programs next to each one. The more you do the better you'll be! Feel free to do others that are not linked as well!

1. The first thing you need to do is install **Netbeans** on your home computer. It is **FREE!**

Click [HERE](#) to download **Netbeans** and then follow the onscreen prompts to install it on your computer!

| Java SE and NetBeans Cobundle (JDK 7u80 and NB 8.0.2) | | |
|---|-----------|--|
| Product / File Description | File Size | Download |
| Linux x86 | 233.68 MB | jdk-7u80-nb-8_0_2-linux-i586.sh |
| Linux x64 | 229.83 MB | jdk-7u80-nb-8_0_2-linux-x64.sh |
| Mac OS X x64 | 301.5 MB | jdk-7u80-nb-8_0_2-macosx-x64.dmg |
| Windows x86 | 243.83 MB | jdk-7u80-nb-8_0_2-windows-i586.exe |
| Windows x64 | 246.65 MB | jdk-7u80-nb-8_0_2-windows-x64.exe |

Once you have **Netbeans** installed it's time to write our first program!



2. Creating your first Java program using Netbeans

How to Output information to a user – follow the Tutorial Video [HERE](#) and then Copy and Paste your program code for this below (or use the Print Screen and then Paste)

```
4  /* and open the template in the editor.
5  */
6  package javaapplication206;
7
8  /**
9   *
10  * @author dbeasley
11  */
12  public class JavaApplication206 {
13
14  /**
15   * @param args the command line arguments
16   */
17  public static void main(String[] args) {
18      System.out.println("Hello you Legend!");
19      // TODO code application logic here
20  }
21  }
22  }
```

Example:

```
System.out.println("I am a legend");
```

3. Using variables to store values in your program – Click [HERE](#) to watch the tutorial on variables and then Copy and Paste your program code for this below

4. Reading User Input to interact with your program – Click [HERE](#) to watch the tutorial on reading User Input and then Copy and Paste your program code for this below

5. Reading in numbers, using Decimals and performing calculations –
Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

6. Maths operators – Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

7. Incrementing variables – Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

8. If statements – Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

9. Logical Operators – Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

10. Nesting Else If– Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

11. While Loop– Click [HERE](#) to watch the tutorial and then Copy and Paste your program code for this below

Programming Challenges!

Following instructions is one thing – but can you show your problem solving skills to create a program for each of the challenges below?

For some of them you may need to view additional tutorials or use Google for help (For each one you complete or attempt, paste your code below the relevant challenge)

1. Create a program that will output the subjects you are studying in Year 12
2. Create a program that will ask the user what colour is their favourite and then print out their choice. E.g. “What is your favourite Colour” ->Black “I see your favourite colour is Black”.
3. Create a program that will ask someone how old they are and then tell them how old they will be in 10 years’ time.
4. Create a program that will ask the user a maths question and then will tell them if they got it right or wrong.
5. Create a program that will ask them repeatedly if they are feeling annoyed. It will keep going until they type “yes”.
6. Create a program that will give the user a quiz. Each question they get right should make their score increase. At the end of the quiz tell them their score! At the end of the quiz ask them if they want to take the quiz again

If you've completed or even attempted all the above, congratulations you are in great shape and well prepared for your Computer Science A Level!