Transition to Advanced Level

Computer Science

Contents:

I. Research Topics

These questions require you to use your technical knowledge in context. You will need to do some research to extend your knowledge and provide relevant examples. Reference any sources that you use to help you – there is a guide to referencing at the bottom of the page.

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 hardware features mainly RAM, CPU and storage capacities of 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
    d. As a tool for education

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

5. There have been many recent high-profile cyber-attacks across the world and some commentators have said that “we now rely too much on technology” .
   Write an essay explaining how far you agree with this statement, including descriptions of threats to IT systems and ways to reduce vulnerabilities.

6. 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
    - Legal issues related the introduction of GDPR
    - 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.

A guide to the Harvard referencing style can be found here:

https://www2.le.ac.uk/library/help/referencing/author-date

There are also many free tools online to help you create references. Make sure that you reference each source carefully and try to use a range of sources, including books and magazines, rather than just websites.

II. Puzzles and Problem-Solving 1: Logic Puzzles

The following puzzles and problems require you to think logically and systematically.

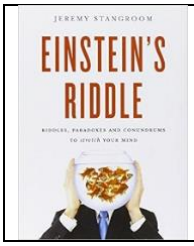
A deduction puzzle

There are four programmers, each of whom codes in a different language and has their own reason for studying computer science. Can you use the clues provided to match each programmer to the correct programming language and motivation for studying computer science?

1. Of the one who likes puzzles and the one who loves maths, one is Alice and the other programs in C.
2. The python programmer's name is alphabetically one more than the person who enjoys solving puzzles
3. Bob got into computer science through gaming
4. Of Dave and Bob, one wants to study computer science for the money, while the other codes in VB

Lots more of this style of puzzle, including interactive solving tools, can be found here: http://www.logic-puzzles.org/index.php

For Einstein's riddle, allegedly one of the hardest of this type of puzzle, try this book:

	Einstein's Riddle Jeremy Stangroom Bloomsbury Publishing (18 May 2009) ISBN-10: 1408801493 ISBN-13: 978-1408801499	Contains the world's most famous logic puzzle

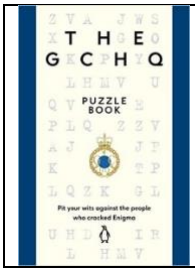
Knights, knaves and spies

On the fabled Island of Knights and Knaves, we meet three people, A, B, and C, one of whom is a knight, one a knave, and one a spy. The knight always tells the truth, the knave always lies, and the spy can either lie or tell the truth.

A says: "C is a knave."
B says: "A is a knight."
C says: "I am the spy."

Who is the knight, who the knave and who the spy?

If you like logic puzzles, code breaking, or just really hard problems, try this:

	The GCHQ Puzzle Book GCHQ Michael Joseph (20 Oct. 2016) ISBN-10: 0718185544 ISBN-13: 978-0718185541	A proper work-out for the brain!

III. Puzzles and Problem-Solving 2: Algorithmic Thinking

Each of the following puzzles requires you to design an algorithm. You may want to start by writing out some examples to help you understand the problem.

Weighing and measuring

1. You have 10 bags of coins, each bag contains 100 coins. Nine of the bags contain real coins; each real coin weighs 1 gram. One bag contains fake coins; each fake coin weighs 0.9 grams.

   If you have an accurate scale that will display the weight of an object placed on it, how can you identify the bag of forgeries using the scale only once?

2. You have 12 coins, one of which is fake. The fake is either lighter or heavier than the real coins, but you do not know which. You have a balance that you can use to compare the weights of items.

   How can you find the fake coin in just three uses of the balance? (You have no other weights or reference objects, just the balance and 12 coins.

Light switches

This puzzle was an Oxford University interview question.

You are standing in a room with three light switches. Each switch controls exactly one light bulb in the next room. (This is a budget puzzle, so they are plain, cheap, basic light bulbs.) The door to the next room is closed, and there are no windows, so you cannot see the light bulbs. You may manipulate the switches as much as you like, then you may go through into the room with the lights. You must then say which switch controls which bulb. How do you do it?

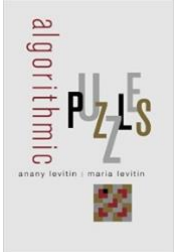
Make 15

You and I are going to play a card game. The rules are as follows:

- 9 cards, numbered 1 – 9, a placed face up on the table between us
- You go first
- On your turn you may pick up any one card from the table
- We alternate turns, each picking up one card at a time
- The winner is the first player to get any three cards that add up to exactly 15 (You can have more than three cards in your hand as long as three of them add up to 15. For example, if I was holding 8, 6, 2 and I could pick up the 5 I would win with 8, 2, 5)

What strategy should you follow to always win at this game, or at least never lose?

Many more of these algorithmic puzzles can be found here:

	Algorithmic Puzzles Anany Levtin, Maria Levtin Oxford University Press, USA (14 Oct. 2011) ISBN-10: 0199740445 ISBN-13: 978-0199740444	A collection of puzzles designed to test and develop your algorithmic thinking and problem solving strategies. The book is well organised, with a discussion of each problem solving strategy and then several puzzles to practice.

IV. Writing Code

Write a program in your preferred programming language to showcase your programming skills. You should aim to demonstrate as many skills as you can, as appropriate to your prior knowledge and experience.

Some ideas for programs

- A game
  - A basic game could be text-based
  - More advanced games may have a GUI and/or an AI to play against
  - The "Make 15" game (above) could be a good starting point
- A program to encipher and/or decipher a text file
  - You may want to research well-known ciphers here: http://simonsingh.net/The_Black_Chamber/index.html
- A useful tool, such as a calculator or countdown timer

Demonstrate as many of the following skills as you can:

GCSE standard:

- ☐ Selection (if-elif-else and/or switch-case)
- ☐ Iteration
  - For loop
  - While loop
- ☐ Validation of user input
- ☐ Lists or arrays
- ☐ File reading/writing
- ☐ Functions

Extension skills:

- ☐ GUI
- ☐ OOP

Document your program carefully. You need to show the purpose of your program and evidence of testing your program.

If you are new to programming, you may want to follow an online course or your teacher may recommend a basic textbook. If you have written code before, consider trying a new programming language. Perhaps write the same program in two different languages and compare your solutions.

For a different coding challenge, work your way through the Little Man Computer examples here: https://peterhigginson.co.uk/LMC/ Follow the HELP link to get started. There is an extended simulator linked from the help page! Can you create the examples from the LMC simulator in the more advanced RISC simulator?

For a mix of advanced problem-solving and programming, try the British Informatics Olympiad: http://www.olympiad.org.uk/problems.html
The Mayan Calendar problem is a good starting point, even if you are new to programming.