TCOLC Sixth Form

Yr11 – 12 Transition Activities

Subject: A Level Computer Science

THE CITY OF LEICESTER COLLEGE

A LEVEL COMPUTER SCIENCE OVERVIEW

EXAM BOARD – OCR

You have made an excellent choice in deciding to study Computer Science at A Level.

Computer Science is a practical subject where students can apply the academic principles learned in the classroom to real-world systems. The aims of this qualification are to enable learners to develop the following:

- An understanding and ability to apply the fundamental principles and concepts of computer science, including **computational thinking** (abstraction, decomposition, pattern recognition) **logic and Boolean logic, algorithms, data representation and object-oriented programming (OOP).**
- The ability to analyse problems in computational terms through practical experience of solving such problems, including writing programs (primarily using the python programming language).
- The capacity to think creatively, analytically, logically and critically.
- Mathematical skills to understand data types and primitive data types. The ability to solve binary arithmetic and floating-point arithmetic and solve logic problems using Boolean algebra.

This pack contains a programme of activities and resources to prepare you to start an A Level in Computer Science in September. It is aimed to be used throughout the remainder of the summer term and over the summer holidays to ensure you are ready to start your course in September.

The pack is divided into some of the key topics you will study in A level Computer Science. There are a range of different activities to do in each topic area. The world of computer science continues to develop at an amazing rate. The challenge for you as a computer scientist is to be able to respond to this everchanging world and to develop the knowledge and skills that will help you to understand technology that hasn't yet been invented!

OUR EXPECTATIONS

The key features of this specification encourage the following:

- Emphasis on problem solving using computers.
- Emphasis on computer programming and algorithms.
- Emphasis on the mathematical skills used to express computational laws and processes, e.g. Boolean algebra/logic and comparison of the complexity of algorithms.
- Theory of computation.
- Fundamentals of data representation.
- Fundamentals of computer organization and architecture.
- Systematic approach to problem solving.
- Learning about data structures. Data in a computer program is organized using a data structure. There are different methods for organizing data. Arrays are a common tool used to organized data when programming.
- Understanding databases and how data is exchanged between different systems.
- Legal, moral, cultural and ethical issues surrounding the use of computers and ethical issues that can or may in the future arise from the use of computers.
- The use of algorithms to describe problems and the measures and methods to determine the efficiency of different algorithms, Big O notation.
- Develop and complete a programming project, choosing project title and problem to be solved and using the agile approach (*agile software development is a process for developing software, allowing you (the developer*) *to create your software program in the way most suited).*

At A-Level computer science students are expected to become more independent with their learning.

Good note taking and presentation of work is an essential skill that is needed for success on an A-Level course. Poorly written notes reveal a lack of understanding and do not provide any chance for revision and review at a later date.

You should begin practicing these skills as you work through this pack during the summer term and holidays.

CHARACTERISTICS OF SUCCESSFUL COMPUTER SCIENCE STUDENTS:

- Having a technical interest in computing and computer programming.
- Staying up to date with current technology news and developments in computing.
- Developing their computer programming skills outside of the classroom to ensure they can program with confidence.
- Engaging in classroom discussion, contributing to the learning environment and using free independent study periods to review and recap their learning.
- Resilient problem solvers, if they are stuck then they make an effort to find out the answer for themselves, through internet based research of by reading a textbook/revision guide related to the topic.
- Are proactive and actively look for ways to expand their knowledge and get better at each stage and topic.

OCR A LEVEL SPECIFICATION AT A GLANCE:

Students must take all three components to be awarded the OCR A Level in Computer Science.

Component 1—Computers Systems (01) Written exam paper (2hrs 30mins) worth 40% of total A Level

- The characteristics of contemporary processors, input, output and storage devices.
- Types of software (operating systems) and the different methodologies used to develop software.
- Data exchange, database concepts, introduction to SQL.
- Networks, network security and threats, HTML and web technologies.
- Data types, Boolean algebra, data structures and algorithms (analysis and design of algorithms).
- Legal, moral, cultural and ethical issues.

Component 2—Algorithms and programming (02) Written exam paper (2 hrs 30mins) worth 40% of total A Level.

- What is meant by computational thinking (thinking abstractly, thinking ahead, thinking procedurally, etc.)
- Problem solving and programming—how computers and programs can be used to solve problems.
- Algorithms and how they can be used to describe and solve problems.

Component 3—Programming project (03) - Non-exam assessment (NEA) worth 20% of total A Level.

- Students are expected to apply the principles of computational thinking to a practical coding programming project.
- The project is designed to be independently chosen by the student and provides them with the flexibility to investigate projects within the diverse field of computer science.
- Students are expected to analyse, design, develop, test, evaluate and document a program written in a suitable programming language.

REVIEW/REVISE

COMPUTATIONAL THINKING (Task time approximately 45-60mins)

Watch this video: https://youtu.be/euFj8D1A1Kw and make notes using the Cornell Notes document.

Read the following:
https://www.cambridgemaths.org/Images/espresso_19_computational_thinking_in_the_classroom.pdf

Answer the following using complete sentences and with examples:

1. What is an algorithm?
2. How can you create and algorithm? And provide an example.
3. What is computational thinking?
4. Why is computational thinking required in the study of computer science?
5. Define the term 'abstraction' and provide an example.
6. What is decomposition and provide an example.

So what actually is 'computational thinking'? You should have a basic understanding from your GCSE years**.** It is the thought processes involved in problem solving, so that the solutions are represented in a form that can be effectively carried out by an information-processing agent, such as a computer. Core concepts involved in computational thinking include: algorithmic thinking - developing a set of instructions or sequence of steps to solve a problem; evaluation - ensuring a solution is fit-for-purpose; decomposition - breaking a problem down into its component parts; abstraction - hiding detail or removing complexity without losing the important detail required to solve a problem; generalisation - finding a general approach to a set of problems.

Creativity is important when applying computational thinking principles to a problem. Programming is a fundamentally creative skill - whether it is used to create a search algorithm, build an app or design a website.

Why is it important in the workplace? So much of modern-day business is about problem solving - whether that's making small improvements to enhance the efficiency of a business or creating breakthrough products and services for consumers. Computational thinking runs through every aspect and function of a modern business. It has become more crucial in the 21st century workplace where so much is now data-driven - analysing consumer behaviour, the movement in financial markets and the performance of public services, like health or policing, are just a few job roles that require individuals to be able to think through problems in a way that a computer could understand.

PRACTICE

PYTHON PROGRAMMING PRACTICE

Python programming is fundamental for A Level computing. You will be required to undertake a large project using a programming language during the second year of the course and this will require a good understanding and ability to develop a program and problem solve your programs successfully.

There will be regular programming tasks set throughout this course, and a good knowledge of programming is essential for both the coursework and exam components.

Python programming Basics

Work your way through these Computer Science Circles interactive python tutorials:

https://cscircles.cemc.uwaterloo.ca/

Before you begin you should create an account on the site so that your progress will be saved. This is a site we will continue to refer to and use during the beginning of the A-level course.

This website takes you from the very basics of writing your first Python program all the way through to some quite challenging concepts. If you are already familiar with python the early sections will be quite easy, but you should still work through all of them.

You should aim to spend 2-3 hours each week progressing through these tasks, making good use of the interactive tools to explore coding in Python and completing all of the tasks in each section. You can see a checklist of all tasks and their completing status in the "My progress" section.

Python problem solving challenges

Once you are secure in some of the basics of programming, you also need to develop some problem-solving skills, and apply your programming knowledge to unfamiliar situations.

The best way to do this is by solving puzzles using code, so here you will find a set of puzzles which begin simple, and can all be tackled using python:

https://projecteuler.net/

Sign up for an account so that you can submit answers and see how far through the puzzles you can get!

WATCH

BBC CLICK

BBC Click is the programme for everyone interested in the internet and computing.

Whether it's e-commerce, new developments and products, or gadgets and games, BBC Click looks at the tools that will revolutionise business and personal life in the future.

This is a really useful website for staying up to date with current technological news and support you in the A level topic concerning ethics, morals, legal and cultural issues surrounding computer science.

Task 1: BBC click review of 2019

This includes space travel, electric cars, 5G and the increased use by police of facial recognition.

Watch the video:

https://www.bbc.co.uk/iplayer/episode/m000d45t/click-best-of-2019

Write an article, summarising one of the topics from the Click review, highlighting one of the aspects they discussed and what were the advantages, or disadvantages of this.

Task 2: Artificial Intelligence (AI) and Coronavirus

The following video looks at how AI tools could help track the virus spread and find new treatments.

Watch the video:

https://www.bbc.co.uk/iplayer/episode/m000g8w5/click-can-ai-help-fight-coronavirus

- You will learn about AI at A Level and will be required to answer an essay question in the exam that may relate to AI.
- Do some further research on AI and see below example exam question and have a go at answering.

Answer the following question:

1. "Developments in Artificial Intelligence mean that in twenty year's time most people will be unemployed." Discuss whether or not you agree with this statement.

LISTEN TO

NETWORK SECURITY AND THREATS

You will discuss and learn about systems security, firewalls and encryption during your A level.

Listen to the following Podcast which talks about security, hacking and university hacking.

Podcast: https://www.smashingsecurity.com/176

Task 1: Write a summary from the **podcast**—what are the key points they discuss? *They talk about some Zoom meeting 'hacks', the Computer Misuse Act, Cybercrime, data breaches at Warwick University and lots more!*

Task 2: Read the following security workbook, do some research on Google around the topics in the workbook and **answer** the questions in the workbook.

Task 3. Watch/listen to the following Ted Talk: "Why study computer science!"

https://youtu.be/t3Y4p-6YWnQ

Answer the following:

1. Why have you chosen the study Computer Science at A Level?
2. What do you aim to achieve from the course?
3. What skills are you aiming to develop?
4. What other A Levels have you chosen and why?
5. What programming languages have you used?

READ

DATA REPRESENTATION

During your GCSE you learnt about binary, denary, hexadecimal, binary arithmetic and ASCII. For A level you will also learn about the following; (some of this is covered in year 2, so don't worry you don't need to know it all now!).

- representing negative numbers using two's complement
- fixed point binary numbers
- floating point arithmetic

Complete the following tasks and then complete the worksheet/test to help remind you of the skills from GCSE or re-learn the basic data representation techniques.

1. Read the following BBC bitesize data representation pages (10 pages)
    a. https://www.bbc.co.uk/bitesize/guides/zfspfcw/revision/1
2. Complete the quiz and make a note of your score:
    a. https://www.bbc.co.uk/bitesize/guides/zfspfcw/test
3. Read the following on data representation
    a. https://www.bbc.co.uk/bitesize/guides/zsnbr82/revision/1
4. Complete the data representation worksheet.

RESEARCH

SYSTEMS ARCHITECTURE

One of the first topics you will start at A level is about the components of a computer. The CPU, also known simply as the processor, has a number of different components which enable it to carry out its task of executing instructions. These components include:

- Control unit (CU)
- System Busses (Control, Data & Address bus)
- ALU (Arithmetic Logic Unit)

Dedicated registers

- Program counter (PC)
- Current instruction register CIR)
- Memory address register (MAR)
- Memory data register (MDR)
- Accumulator (ACC)

The sequencing of these instructions can be divided into 3 phrases using the:

Fetch—**Decode**—**Execute** cycle.

Tasks:

1. Read about the history of computing. Choosing at least 6 historic computing moments, create a time-line explaining each of these and the impact it has had on computing today.

https://www.computerhistory.org/timeline/computers/

2. Research each of the keywords above (components and registers) and **create a revision guide or poster,** which explains each register and component.

ADDITIONAL RESOURCES/LINKS

Read/learn—You can use the full BBC Bitesize OCR GCSE course (read and do the tests) to recap your knowledge to prepare you for A level computer science.
https://www.bbc.co.uk/bitesize/examspecs/zmtchbk

Watch the Alan Turing movie—The Imitation Game. Alan Turing cracked codes produced by the German military's seemingly unbreakable Enigma machine during World War II using math, engineering and still-to-be-invented computer science.
https://www.rottentomatoes.com/m/the_imitation_game

Read/puzzles —CS4FN (Computer Science for Fun) is a magazine on computer science aimed at school students "Explore how computer science is also about people, solving puzzles, creativity, changing the future and, most of all, having fun." It is printed twice a year and has an associated website with additional articles.
http://www.cs4fn.org

TEST/Quizzes – TestandTrack, it provides a 1,000+ interactive quizzes to facilitate learning. Tests cover all computer science and programming topics, including GCSE, A Level.
https://www.testandtrack.io/index.php/studenttest/test

News—Wired. WIRED IS WHERE tomorrow is realised. It is the essential source of information and ideas that make sense of a world in constant transformation. The WIRED conversation illuminates how technology is changing every aspect of our lives—from culture to business, science to design.
https://www.wired.com/tag/computer-science/

Programming—SoloLearn Python and Java—great free website to improve your skills and knowledge. https://www.sololearn.com/Course/Python/

Programming—Automate the Boring Stuff with Python is a fantastic practical course on using Python to automate processes one your computer, there is also a YouTube playlist which talks through each chapter
https://automatetheboringstuff.com/
https://www.youtube.com/playlist?list=PLGoJzB271_7r-iLYuEHEPJ5pSIYxXjJEn

The Alan Turing Institute—"We believe data science and artificial intelligence will change the world." https://www.turing.ac.uk

Careers —If you go on to further study Computer Science at Degree level or apprenticeships, here are some links for various opportunities:
https://www.prospects.ac.uk/careers-advice/what-can-i-do-with-my-degree/computer-science

Apprentice opportunities at GCHQ. GCHQ's unique CyberFirst Degree Apprenticeship is a great opportunity to develop advanced cyber security skills, while helping the UK to grow its technical capabilities.
https://recruitmentservices.applicationtrack.com/vx/lang-en-GB/mobile-0/appcentre-3/brand-4/xf-bd8ab30ccf84/candidate/so/pm/1/pl/6/opp/1876

Apprenticeships at MI5—become a technical apprentice "You'll learn some incredible skills. From creating new apps that track terror suspects, to developing innovative ways to stop espionage in its tracks." https://www.mi5.gov.uk/careers/icanatmi5?adhoc_referrer=011950192003