Sixth Form Handbook

Computer Science

Contents

About the course

The A-Level Computer Science course offered at Poole High is varied, stimulating, thought-provoking and edifying. Taught by teachers who themselves are skilled programmers and former industry professionals, you really will be taught by some of the most experienced post-16 subject specialists in the local area.

At Poole High School we focus on the programming language C#, one of the most widely respected and popular programming languages used by UK enterprise. Your first year will develop your programming skills in C# and you will learn how to apply your programming knowledge to solving real-world problems. If you previously studied Computer Science at a lower level, you will recognize the basics of many topics such as sequencing, selection and iteration; however you will be pushed and extended to topics such as object oriented and functional programming paradigms.

The technical topics covered during the course are wide ranging, and cover themes such as Turing machines, Network architecture, Boolean algebra, managing large data, and mathematics for Computer Science. During your second year, you will also have the chance to construct a computer program on a topic area which interests you as part of your final A-Level assessment.

There is a shortage of programmers in the UK; therefore the possibilities are great for students with A-Levels and degrees in Computer Science - both in this country and abroad. Along with the of computer programming you may find opportunities in computer networking, computer architecture, data management and technical engineering fields.

How you will be assessed

A Level

Paper 1

This paper tests a student's ability to program, as well as their theoretical knowledge of computer science from subject content relating to the fundamentals of programming, data structures, algorithms and the theory of computation.

On-screen exam: 2 hours 30 minutes 40% of A-level

Paper 2

This paper tests a student's ability to answer questions from subject content relating to fundamentals of data representation, computer systems (including organisation and architecture), communication & networking, databases, big data and functional programming as well as management of Big Data and the consequences of uses of computing.

Written exam: 2 hours 30 minutes 40% of A-level

Project:

The non-exam assessment assesses student's ability to use the knowledge and skills gained through the course to solve or investigate a practical problem.

Students will be expected to follow a systematic approach to problem solving. 20% of A-level

Course Reading List & Materials

Books

AQA A-level Computer Science Computing by Bob Reeves, published by Hodder Education ISBN 978-1-471-83951-1

A level Computing by Peter Kemp, available on WikiBooks: Wikibooks - AQA A-level Computing

Useful websites

A-level Computing/AQA wikibook covering the latest specification. This can be printed from the website or accessed online.
https://en.wikibooks.org/wiki/A-level_Computing/AQA

Queen Mary College, London university site of CS4FN (Computer Science for Fun). This site contains wonderful ideas for generating interesting lessons: CS4FN
http://www.cs4fn.org/

To practice assembly language, there is an emulator which you can use:
http://www.peterhigginson.co.uk/AQA/

To practice functional programming, you can use Haskell online:
http://rextester.com/l/haskell_online_compiler

For wonderful animations covering most of the course, visit: Virginia Tech - Animations to Assist Learning Some Key Computer Science Topics
http://courses.cs.vt.edu/~csonline

A course that maps onto the new specification in a useful way. Use this for its applets and commentaries for many aspects of the course: Virginia Tech - Introduction to Computer Science
http://courses.cs.vt.edu/~cs1104/FrontEnd/index.html

Processor simulator (excellent): Brittunculi - the JASP toolkit
http://www.brittunculi.com/jasp/

Slides supporting teaching of machine architecture: Brittunculi - Fundamentals of Computer Architecture
http://www.brittunculi.com/foca/materials

Finite state machines and programming

Use resources from the Swiss Education centre for graph modelling for Finite State Machines, Turing Machines, graphs and graph algorithms, exorciser for computing theory: swisseduc.ch - Computer Science
http://www.swisseduc.ch/compscience

Finite State Machine and Turing Machine simulator (excellent resource): swisseduc.ch - Learn programming with Kara
http://www.swisseduc.ch/compscience/karatojava/kara

swisseduc.ch - Exorciser has another FSM simulator and more. The following is a URL of a site produced by a teacher, Clive Hirst. He has used the resource from the above Swiss site and started to produce class exercises. Clive has agreed to the details of his site being made public: asiplease - Kara Home Page.

http://www.asiplease.net/computing/kara/index.htm

Use ASMTutor from Educational Computing Services limited for an introduction to assembly language. Use GraphTutor from Educational Computing Services limited for an introduction to representing graphs (available December 2008). Use Finite State Simulator from Educational Computing Services limited for simulator for finite state machines.

http://www.educational-computing.co.uk/

Finite State Machine simulator, Turing machine simulator, regular expressions: Duke Universty - JFLAP software

http://www.cs.duke.edu/csed/jflap

Robotics

ipre - Institute for Personal Robots in Education

http://www.roboteducation.org/

ipre - Learning Computing With Robots

http://wiki.roboteducation.org/Introduction_to_Computer_Science_via_Robots

A.I Hub - Jonathan Schaeffer On Artificial Intelligence and Games

http://aihub.net/jonathan-schaeffer-artificial-intelligence-games

Problem solving

Introduction to problem solving with computers: Dave Moursund, University of Oregon - brief introduction to problem solving

http://pages.uoregon.edu/moursund/dave/Article&Presentations/problem-solving.htm

Role of variables

University of Eastern Finland - role of variables

http://cs.joensuu.fi/~saja/var_roles

Tasks before September

At Sixth Form, we program in a language called C#. The first two tasks will get you up and running quickly ahead of your first lesson in September.

1. Download and Install Microsoft Visual Studio Community 2017 (C#) on to your home Computer www.visualstudio.com (Apple and Linux users can use MonoDevelop - http://www.monodevelop.com/). Both programs are free.

2. Complete the first three tutorials for creating your first C# program from https://channel9.msdn.com/Series/C-Fundamentals-for-Absolute-Beginners (You can do more of tutorials, you don't have to stop at three).

Research the answer to the following questions:

3. What is a high level imperative language?

4. Describe the difference between the Bubble Sort and the Insertion Sort algorithms. You should describe this with a diagram.

5. What is git? What is the purpose of the www.github.com online service? How could you use this for your A-Level studies?

6. Read through the course wikibook. This will be an invaluable resource for your A-Level Computer Science studies. https://en.wikibooks.org/wiki/A-level_Computing/AQA

Who can I contact for help?

Mr. P. Thynne (p.thynne@poolehigh.poole.sch.uk)
Mr. K. Lean (k.lean@poolehigh.poole.sch.uk)

In addition – look out for the new Google Classroom: Year 12A-Level Computer Science 2020 intake (code: mmnlf7a).