



Course Outline

ICT Year 9

Inspiring excellence, empowering global minds

Overview

This course is designed to expand student's computational logical thinking skills in programming. To achieve this student's will be using python as their programming language. Throughout the course students will learn the fundamentals of programming and how they can be used together to create a diverse and complex computer program.

Learning Outcomes

The aims of the ICT Syllabus are to encourage and enable students to:

- To gain a better understanding of the uses of programming.
- Acquire knowledge on how different programming techniques work.
- Become fluent in the basics of programming.
- To gain experience in a programming language
- Become competent in creating simple programs.
- Learn how logical thinking works.
- Become accustomed to debugging programs.
- Use research to expand on programming knowledge.
- Carry out discussions to express their thoughts on specific programming techniques.
- Design and manipulate webpages
- E Safety -risks and Measures
- Create presentations for Audience considering different factors

Ongoing Objectives

Theoretical concepts

- To learn what logical thinking is and how to think in a logical manner.
- Learn what sequencing is and the importance of sequencing when programming.
- Know the purpose of using different looks in different circumstances.
- Learn how a program can be expanded to make it more complex
- To articulate and justify answers, arguments and opinions
- To learn advantages and disadvantages of different techniques

- To maintain attention and participate actively in collaborative conversations, staying on topic and initiating and responding to comments
- Continue to grow and expand on the use of key terminology.
- Learn why different programming skills may be used.
- Understand when to use different programming aspects such as variables, lists, loops or user defined function etc.

Programming

- To use relevant strategies to build their practical skills
- Expand knowledge and competence in using python
- Learn how to create variable, lists, function, statements, loops, user defined functions etc.
- Select appropriate techniques needed for a certain task
- To learn how different techniques can be combined to simplify code
- Learn how to structure code in a more efficient manner
- Debug programs and getting rid of redundant code
- Understand the benefit of creating user defined functions and the benefits of learning programming in the real world.

Unit Overviews

Term 1

Chapter 2 - Variables

Approximate length: 2 weeks

In the first section of this chapter, students will be taught about variables. Students will be taught the skills required to create an effective variable along with why and how they may be used. Skills to be taught are Assigning a variable, printing a variable, using numbers, strings and lists in a variable.

Specific National Curriculum Objectives Covered:

- To learn the basic fundamentals of programming
- To use their second programming language
- To start thinking with computational logic in mind
- To have an understanding of the use and importance of using variables in every program
- To understand the versatility of variables and how they can be used with other functions such as strings, numbers and lists etc.

Chapter 2 – Making decisions**Approximate length: 2 weeks**

This section of the chapter will focus on how students can use if and else statements to allow the program to make predefined decisions in the program. Students will be learning skills that show them how to use Boolean values, Boolean expressions and how they can use branches to branch different if and else statements.

Specific National Curriculum Objectives Covered:

- To learn the basic fundamentals of programming
- To further understand computational logic using Boolean (True/False)
- To start using simple Boolean operations to compare different statements
- To know how different Boolean operations work
- To be able to incorporate Boolean with if and else statements

Chapter 2 – Loopy loops**Approximate length: 2 weeks**

The loopy loops part of the chapter will allow students to learn how different loops can be created and each of the loops allows them to do. We will focus on two main loops: for loops, while loops. The chapter will also delve deeper into how students can create an infinite loop and how to use a loop within a loop.

Specific National Curriculum Objectives Covered:

- To learn the basic fundamentals of programming
- To learn how programs can repeat tasks based on the outcome required
- To know the difference between the different types of loops
- To be able to make a logical decision when to use the different types of loops
- To know how to incorporate a loop within a loop

Chapter 2 – Functions**Approximate length: 2 weeks**

Students will be taught about the two different types of functions: Built in function and user defined functions. Students will be shown different types of built in functions that they have been using and then be shown why it's important to create their own functions to carry out different tasks as this will reduce redundant code.

Specific National Curriculum Objectives Covered:

- To correctly use the different built in functions
- To think logically and use the correct function based on the task
- To understand the differences between built in function and user defined functions
- To create a simple user defined function
- To know how to run a user defined function multiple times
- To understand the benefits of creating a user defined function compared to just writing the code for a task.

Chapter 3 – Robot builder**Approximate length: 2 weeks**

Chapter 3 moves onto teaching students how to create drawings in python using a module called turtle graphics along with other skills taught in chapter 2. The first section focusses getting students to draw simple shapes

such as squares and rectangles. Students will need to make different sized shapes and place them in different locations using the goto function.

Specific National Curriculum Objectives Covered:

- To use an external module (Turtle)
- To learn the functions within turtle graphics
- To create simple shapes such as squares and rectangles
- To be able to add color to the shapes created
- To use the goto function to place shapes in different locations to make a robot.

Term 2

HTML

Approximate length: 4 weeks

This topic focusses on students learning about hypertext markup language (HTML). The main focus of this chapter will be teaching students how to create a HTML webpage, create tables, add colored background, inserting images, creating tables, adding different types of text such as paragraphs, headings, body text and learning about tags.

Specific National Curriculum Objectives Covered:

- To learn how to create a HTML webpage
- To continue developing understanding of programming languages
- To learn how to add elements in a HTML webpage
- To combine different skills together to make a unique and rich website
- Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users.
- Develop their capability, creativity and knowledge in computer science, digital media and information technology.

Term 3

Presentations

Approximate length: 3 weeks

In this chapter, students will learn practical skills on how they can create a professional presentation. The chapter will focus on key skills such as master slides, animations, importing videos and pictures, using slide templates and how to use notes. All these skills will develop students understanding of how the software works and how they can use the skills to present themselves.

Specific National Curriculum Objectives Covered:

- To develop practical skills in creating presentations using professional software
- To understand the different applications of an effective presentation

- To be able to use a variety of different software's
- To know how to create master slides to give a presentation a consistent look.
- To incorporate different features to create an interactive presentation

E-safety

Approximate length: 1 week

In this unit, the students explore and explain why personal data should be kept confidential, describe how personal data can be gathered by unauthorized persons (including: by smishing, vishing, phishing and pharming), and how this might be prevented, discuss why e-safety is necessary, describe malware issues (including: Trojan Horse, worms, spyware, adware, rootkit, malicious bots, ransomware).

Specific National Curriculum Objectives Covered:

All pupils must have the opportunity to study aspects of information technology and computer science at sufficient depth to allow them to progress to higher levels of study or to a professional career.

All pupils should be taught to:

- understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to identify and report a range of concerns.

Introduction to computer systems

Approximate length: 3 weeks

This topic will be focusing on getting students introduced to the basics of a computer system. Students will learn about the different types of computer systems along with input and output devices. They will also learn about different hardware and software requirements of a computer system.

Specific National Curriculum Objectives Covered:

- To learn what makes a computer system
- To develop understanding on how different components are linked together
- To consolidate understanding of the need for both hardware and software
- To be able to differentiate between input and output devices

Assessment

Formative: Throughout the units, the students will complete graded work, quizzes and programming activities which allows the teacher to assess the students attainment and inform their planning.

For each unit the students complete a pre and post quiz. This allows us to see progress across the units.

Summative: At the end of each term, students will complete standardized internal tests. This allows us to measure the students' progress throughout the term and year. At the end of the academic year, the students complete an open book project where their programming skills and use of different techniques will be tested.

Remote learning: Live lessons will be carried out using Microsoft Team, Phoenix or Zoom. Students will also be provided with prerecorded tutorials which they can access at anytime and this will help

assist the lower ability students along with videos demonstrating step by step procedures solving practical concepts.

Remote learning: One to one questioning and group questioning will be carried out. Online quizzes will be carried out using Microsoft forms. Students will also be asked to create different projects using a range of software's.