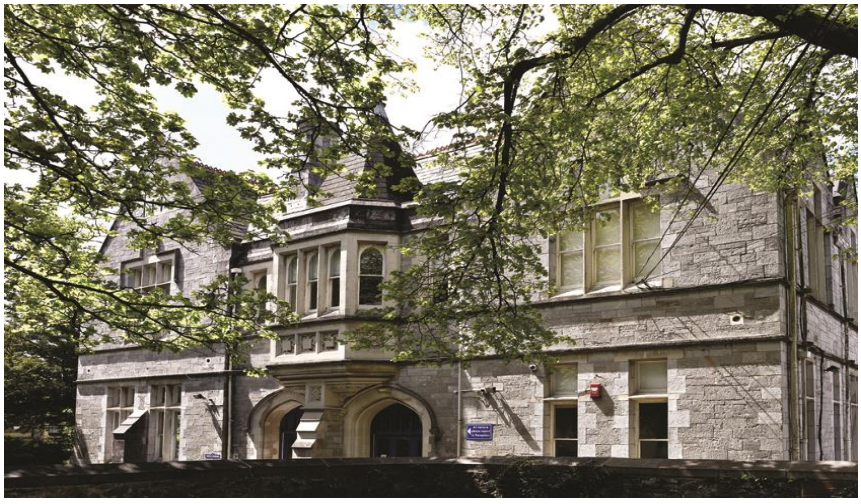
Curriculum and Progression Overview

Computing

Plymouth High School for Girls



CONTENTS

1. Curriculum

Plymouth High School for Girls aims to be an outstanding Grammar School for the 21st century remaining true to the philosophy of the school's motto 'For Life Not School We Learn'.

A school which:

- Promotes a culture of high aspiration enabling personal and academic success
- Prepares students to think confidently and independently
- Provides an outstanding education for all, the key to this being high quality teaching
- Provides enjoyment, excitement and challenge for all, stimulating an enthusiasm for lifelong learning
- Prepares students to become active citizens, equipped to succeed in a world of rapid change

Computers are now part of everyday life and, for most of us, technology is essential to our lives, at home and at work. 'Computational thinking' is a skill that all pupils must learn if they are to be ready for modern the workplace and able to participate and develop effectively in the digital world.

Computer Science is an engaging and practical subject, encouraging creativity and problem solving. It encourages students to develop their thinking skills, understanding and application of the core concepts in computer science. Students also analyse problems in computational terms and devise creative solutions by designing, writing, testing and evaluating programs.

Rationale

Through IT and Computing in KS3 and KS4, students will learn how computers and computer systems work, they will design and build programs, they will develop their ideas using technology, create a range of digital content and solve problems.

The three main strands of computing are:

- Computer Science
- Digital Literacy
- Information Technology.

Computer science is the scientific and practical study of computation: what can be computed, how to compute it, and how computation may be applied to the solution of problems.

Digital literacy is the ability to effectively, responsibly, safely and critically navigate, evaluate and create digital artefacts using a range of digital technologies. The creation of digital artefacts will be

integral to much of the learning of computing. Digital artefacts can take many forms, including digital images, computer programs, spread sheets and 3D animations.

Information technology is concerned with how computers and telecommunications equipment work, and how they may be applied to the storage, retrieval, transmission and manipulation of data.

Ambition

We want our students to experience success in computer science and develop the confidence to take risks to solve problems, ask questions and explore alternative solutions without fear. They will therefore, enjoy exploring and applying concepts to understand and solve problems, explaining their thinking and presenting their solutions to others in a variety of ways.

At all stages, the use of collaborative learning and thinking skills encourage students to reason logically and creatively through discussion, and the sharing of ideas and concepts. Misconceptions and wrong answers are an opportunity to improve and deepen understanding of computational concepts, through use of effective questioning and discussion.

We want students to :

- To develop a positive attitude and thinking approach to computing as an interesting and exciting subject in which all students gain success and enjoyment
- To develop computational understanding through systematic direct teaching of appropriate learning objectives
- To encourage the effective use of computing as a tool in a wide range of activities within school and, subsequently, adult life
- To develop an ability in students to express themselves fluently, to talk about the subject with confidence, using correct computational language and vocabulary
- To develop and make connections within computing
- To develop the ability to think clearly and logically with independence of thought and flexibility of mind
- To develop computational skills and knowledge and a quick recall of basic facts

Implementation

The National Curriculum is taught in Year 7,8 and 9, however if there are overlaps or opportunities to stretch and challenge students by introducing elements from the GCSE specifications, these may be taken. GCSE option subjects are chosen during Year 9.

The aims of computing as a whole also reflect the distinction between the three strands.

- Students can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation (CS)
- Students can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems (CS)
- Students can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems (IT)

- Students are responsible, competent, confident and creative users of information and communication technology. (DL)

The strands above are implemented through:

- Planned active learning with opportunities to observe, think, explore, investigate, experiment and play
- Development of problem-solving capabilities, with an increased opportunity for discussion, communication and explanation of thinking (think, pair, share)
- Plan for both depth and breadth with topics as well as across the curriculum as a whole.
- Use of relevant contexts, familiar to young people's experiences
- Collaborative and independent learning using thinking tools
- Making links across the curriculum where appropriate
- Sharing of resources across the department to encourage equitable delivery and team planning to take place wherever possible

Impact

In computing, as in other curricular areas, assessment forms an integral and on-going part of learning and teaching. It is the gathering of evidence of both attainment and progression for the purpose of informing future planning and interventions of teaching. In working with students, our teachers continuously evaluate and make use of this assessment in planning future activities.

- To share learning objectives and success criteria clearly with students
- To assess understanding through skilful questioning within lessons
- To give students clear and regular feedback in both verbal and written form
- To assist with the identifying of the next steps in the learning process which will ensure progression
- To engage in the process of self/peer assessment including retrieval
- To encourage students to use thinking skills in their approach to problem solving

At the end of each unit feedback is sought from the students on WWW and EBI and teachers where appropriate act on the suggestions/comments. Staff also review the SOL, the objectives, the tasks and the outcomes of students to help further develop an inclusive yet challenging curriculum.

Please see the following link for further enrichment material:

https://csunplugged.org/en/ https://code.org/

2. CURRICULUM & ASSESSMENT OVERVIEW: KS 3 Computer Science

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
KEY STAGE 3						
Year 7	*Topic(s): Visual programming (KODU)* *Assessment: Finished game with skills applied and assessment &homework*	*Topic(s): Touch Typing/ Computing Key Skills* *Assessment: Progression test with timed assessment &homework*	*Topic(s): Webpage creation using HTML code* *Assessment: Finished website with skills applied and assessment &homework*	*Topic(s): How computers work* *Assessment: Finished topics with skills applied and assessment&homework*	*Topic(s): Microsoft office higher level Word and PowerPoint skills* *Assessment: Finished topics with skills applied and assessment &homework*	*Topic(s): Microsoft office higher level Excel skills* *Assessment: Finished topics with skills applied and assessment &homework*
Year 8	*Topic(s): App design and implementation* *Assessment: Finished App with skills applied and assessment &homework*	*Topic(s): Touch Typing (intermediate level)* *Assessment: Progression test with timed assessment &homework*	*Topic(s): Introduction into Python programming* *Assessment: Programming assessment &homework*	*Topic(s): Computing Hardware/Software* *Assessment: Finished topics with skills applied and assessment &homework*	*Topic(s): Office key skills project* *Assessment: Finished topics with skills applied and assessment &homework*	*Topic(s): Blender graphics* *Assessment: Finished topics with skills applied and assessment &homework*
Year 9	*Topic(s): Google Sketchup 3D modelling* *Assessment: Finished model with skills applied*	*Topic(s): Micro;Bit Pet project* *Assessment: Finished Pet with skills applied and assessment &homework*	*Topic(s): Cyber security* *Assessment: Finished topics with skills applied and assessment &homework*	*Topic(s): Entry level computer science* *Assessment:* Hardware, Software and Logic assessments *&homework*	*Topic(s): Office key skills qualification* *Assessment: Finished project with skills applied and assessment*	*Topic(s): Programming project* *Assessment:* Practice controlled assessment for

	and assessment &homework				*(AWARDED) &homework*	computer science *&homework*

CURRICULUM & ASSESSMENT OVERVIEW: *Computer Science*

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
KEY STAGE 4						
Year 10	*Topic(s):* Systems Architecture Memory *Programming* *Assessment:* Test and working solutions to programming problems *&homework*	*Topic(s):* Storage Wired and Wireless networks *Programming* *Assessment: Test* and working solutions to programming problems *&homework*	*Topic(s):* Network Topologies, Protocols and Layers Systems Security *Programming* *Assessment: Test* and working solutions to programming problems *&homework*	*Topic(s):* Systems Software Ethical, Legal, Cultural and Environmental concerns *Programming* *Assessment: Test* and working solutions to programming problems *&homework*	*Topic(s):* *Programming Project* *Assessment: Working solution to the programming project &homework*	*Topic(s):* *Programming Project* *MOCK EXAM (OR Exemplar Assessments) &homework*
Year 11	*Topic(s):* Algorithms Programming Techniques *Assessment: Working solutions to*	*Topic(s):* Producing robust programs *Computational Logic* *Assessment: Working solutions to*	*Topic(s):* Translators and Facilities of the Language Data Representation *Assessment: Test and Working solutions to*	*Topic(s):* *Revision* *Assessment: Various tests &homework*	*Topic(s):* *Revision* *Assessment: Various tests &homework*	**GCSE EXAMS**

	programming problems &homework	*programming problems &homework*	*programming problems &homework*			

2. PROGRESSION IN KS3: COMPUTER SCIENCE

	Computer Science	Information Technology	Digital Literacy
Grades 7-9	Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems Use logical reasoning to compare the utility of alternative algorithms for the same problem. Understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits Design and use computational abstractions that model the state and behaviour of real-world problems and physical systems Use two programming languages, to solve a variety of computational problems; Make use of data structures [for example, lists, tables or arrays];	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 create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems Recognise inappropriate content, contact and conduct and know how to report concerns. Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy;
Grades 5-7-	Use computational abstractions that model the state and behaviour of real-world problems and physical systems Use one programming language, to solve a variety of computational problems;	Undertake creative projects that involve selecting, using, and combining multiple applications, including collecting and analysing data	Understand the hardware and software components that make up computer systems Understand a range of ways to use technology safely, respectfully, responsibly and securely

	Understand how numbers can be represented in binary, Solve problems by decomposing them into smaller parts Work with variables Use logical reasoning to explain how some simple algorithms work Use logical reasoning to detect and correct errors in algorithms	Re-use and revise digital artefacts for a given audience, with attention to trustworthiness, design and usability Combine a variety of software to accomplish given goals Select, use and combine software on a range of digital devices Analyse data Evaluate data Design and create systems	Understand the opportunities computer networks offer for collaboration Be discerning in evaluating digital content
Grades 3-4	Design programs that accomplish specific goals Design and create programs Debug programs that accomplish specific goals Use repetition in programs Control or simulate physical systems Use logical reasoning to detect and correct errors in programs Understand how computer networks can provide multiple services, such as the World Wide Web Write programs that accomplish specific goals Work with various forms of input Work with various forms of output	Select a variety of software to accomplish given goals Select, use and combine internet services Analyse information Evaluation information Collect data Present data Use search technologies effectively Use a variety of software to accomplish given goals Collect information Design and create content Present information	Understand the opportunities computer networks offer for communication Identify a range of ways to report concerns about content Recognise acceptable/unacceptable behaviour Use technology responsibly Identify a range of ways to report concerns about contact

Grades 1-2	Understand that algorithms are implemented as programs on digital devices Understand that programs execute by following precise and unambiguous instructions Debug simple programs Use logical reasoning to predict the behaviour of simple programs Understand what algorithms are Create simple programs	Use technology purposefully to organise digital content Use technology purposefully to manipulate digital content Use technology purposefully to create digital content Use technology purposefully to store digital content Use technology purposefully to retrieve digital content	Use technology respectfully Identify where to go for help and support when they have concerns about content or contact on the internet to other online technologies Use technology safely Keep personal information private Recognise common uses of information technology beyond school

3. FEEDBACK AND ASSESSMENT SYSTEM

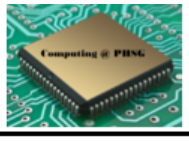
Here is an example of the feedback system used in IT & Computing.

Feedback is given on an individual basis and recorded on the student's personal objective sheet electronically. At the start of each half term they will be given a student friendly version of the half term about to be undertaken.

Students are encouraged to use the objective sheet to self-monitor and keep track of their level by selecting when they understand each topic. At the end of the half term there is a clear view of progression.

As students assess themselves and change the "smiley face", a specific target appears for that objective that will suggest ways for them to make further progress from Emerging to Securing.

YEAR 7 FEEDBACK SHEET

Name			16

Level	Objective	Understanding	
1	Add terrain, heights and a variety of colours and a pond to your world	yes	
2	Add objects to your worldsuch as a fish to your world	yes	
3	Add a kodu to your world and program it to move	yes	
4	Program the fish to move along a given path in the pond	yes	
5	Program the Kodu to eat the fish in the pond	yes	
6	Add a scoreboard to your world	yes	
7	Plan first, and then create your own game using the skills learnt	yes	
8	When a certain amount of points have been gained you win the game	yes	

YEAR 9 FEEDBACK SHEET

Name			28

Level	Objective	Understanding	
1	Revision - Output to the screen / Storing data in variables	yes	
2	Inputting data / Calculations	yes	
3	Data types / Selection with IF	yes	
4	Making programs easier to read	yes	
5	Iteration - Count / Condition controlled loops	yes	
6-7	Subroutines: Procedures and Functions	yes	
7-8	Lists / String handling	yes	
9	Reading and writing to files	yes