

|  |  |  |  |
| --- | --- | --- | --- |
| **Computer Science**  | **Key Stage 3** | **Year 7/8** | **Autumn Term**  |





|  |  |  |
| --- | --- | --- |
| **Key Concepts and Processes** | **Expectations and Assessment Objectives** It is hoped that this scheme of work will provide students with the foundation to develop their ability to use logic and computational thinking to solve a range of problems, whilst giving due regards to the ethics of their approach. It is hoped that students will develop a solid grounding to be able to progress to KS4 with confidence. | **Cross Curricular and Interdisciplinary Links** |
| Computer SystemsData StructuresAlgorithmsProgrammingNetwork,Communicationand Internet | **All** will **Most** will **Some** will  | **Research Projects:** Submission - First week of Spring term.**RP\_1**Intro to computers - ResearchHistory of Computer (abacus > room > desktop > phone <http://www.bbc.co.uk/learningzone/clips/12724.html>Computer history Museum [website](http://www.computerhistory.org/) **RP\_2** Design a A3/A2 poster which contain detail information about one of the following topics. You must be succinct in your details , include all the necessary information. Presentation does count.1. Binary numbers : Why we need binary numbers, what are they, how to add/subtract, multiply binary 2. Data types: Various data types used in programming, what are they? why we need them. 3. Design your own computer, explain how will it work, what are the input, output devices. Be creative your architecture could be different from Von Neumann. Include research work, your ideas, pictures, tables, graphs etc.**RP\_3**Research online on one of the following topics. Write a detailed report upto 350 words. You can include pictures or drawing to make it more presentable.What is Cloud Computing? or What is Artificial intelligence? read on Artificial intelligence [here](http://www.cs4fn.org/ai/) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Learning objectives**  | **Differentiated teaching points/activities** including stretch and challenge | **Suggested resources** | **Suggested AfL activities** and formal assessment (when appropriate)/ Home learning | **Keywords** |
| Week 1 | Understand the Von Neumann Architecture. Name different part of a Computer system. Input devices, output devices, CPU, ALU, storage .etc AT5 | * **All** – Should be able to name different parts of a computer system.
* **Most** – Should be able to understand the Von Neumann Architecture
* **Some** – will be able to identify various peripheral devices including the types of storagre devices.
 | * A PPT on [Von Neumann Architecture](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNcV84a2wxZ3dWZ1E)

* Website for further reading on [Von Neumann Architecture](http://www.teach-ict.com/as_as_computing/ocr/H447/F453/3_3_3/vonn_neuman/miniweb/pg3.htm)
 | * **In formal assessment:**

Draw the Von Neumann architecture of your chosen device. label and name each part clearly | Von Neumann Architecture, Central processing unit, Arithmetic and logic unit, motherboard, processor, input devices, output devices, ROM , RAM etc |
| Week 2 | Understand and differentiate between the hardware and software components that make up computer systems. Understand the role of various hardware components and how they communicate with one another and with other systems. AT5 | * **All** – Computers are devices for executing programs and they use stored sequences of instructions.
* **Most** – Computers consist of number of hardware components each with a specific role like CPU,RAM,ROM,motherboard, powersupply etc
* **Some** – Moore's Law: "Computers are very fast, and getting faster all the time..."
 | * [Inside a Computer](http://www.gcflearnfree.org/computerbasics/7)
* Computer tour a video showing computer parts [here](http://computer.howstuffworks.com/23-computer-tour-video.htm)
 | * **Home Learning**- Research on internet to find the name of different non-volatile secondary storage devices.
 | Hardware, software, CPU, ALU, motherboard, RAM, ROM, Cache, volatile memory, non-volatile memory, memory card, secondary storage, peripheral devices.Floppy drive, Flalsh drive |
| Week 3 | Intro to Binary numbers. Know what is a bit, byte etc . Read and Write numbers in binary number system. Create a base 2 binary table for a byte. AT4 | * **All** – Understand that computers can only interpret and store binary numbers.know what is a bit , nibble and byte.
* **Most** – will read and write numbers in binary systems
* **Some** – Know octal or hexadecimal base system
 | * Intro to binary [here](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNeGpmaFNFcHdyakE)
* Maths is fun A [website](http://www.mathsisfun.com/binary-number-system.html) with lots of information on Binary numbers
* [Binary Piano](http://www.mathmaniacs.org/lessons/01-binary/Binary_Piano/) a fun activity.
* A very interesting way to practise writing binary numbers [online game](http://forums.cisco.com/CertCom/game/binary_game_page.htm)
* Binary [table base 2](https://docs.google.com/file/d/0BxHHYOMUTUwNRS1tTHJnak9EclE/edit)
* Link to worksheets on [binary/octal/hexadecimal](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNenQ4RXhLNGFubEk)
 | * **Home Learning:** Write your name in binary numbers.
* Play the binary game [here](http://britton.disted.camosun.bc.ca/binary.swf)
 | Binary number, denary number, Hexadecimal numbers, bit, byte, nibble, base 2 numbers,, base 10 , base 16 , conversion. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Learning objectives** – including literacy, numeracy and habits of mind (as appropriate) | **Differentiated teaching points/activities** including stretch and challenge | **Suggested resources** | **Suggested AfL activities** and formal assessment (when appropriate) | **Keywords** |
| Week 4 | Writing names using binary, Converting denary to binary and vice versa. Know the American Standard code of information interchange ASCII and why we use it.. AT4 | * **All** – Be able to read and write decimal numbers in binary base 2 system.
* **Most** – Be able to convert decimal to binary and binary to decimal.
* **Some** – Be able to carry out simple operations on binary numbers [for example,

 binary addition. | * PGOnline Lesson plan with resources [Here](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNWmp1bm5pV0NleUE)
* A Website for [Binary Maths](http://www.binarymath.info/addition-subtraction.php)
* resources on [Binary Addition](https://docs.google.com/file/d/0BxHHYOMUTUwNMkJCMFdKQjhUWDg/edit)
* Scratch game on [Binary](http://scratch.mit.edu/projects/dusseau/723745)counting
* ASCII Code [here](http://www.ascii-code.com/)
* More on [ASCII code](http://computer.howstuffworks.com/bytes2.htm)
 | * Assessment :

 Binary and denary number Conversion worksheet [here](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNeGpmaFNFcHdyakE) | * ASCII (American standard code of information interchange), hexadecimal, base, exponent, addition , carry, multiplication.
 |
| Week 5 | Carry out simple operations on binary numbers [for example,binary addition, and binary multiplication] . Create a file in notepad and check the size of file in numbers of bytes by counting the number of characters in the file. AT4 | * **All** – understand that computer save one byte for every character. Check the size of notepad file. (require computers)
* **Most** – Carry out simple addition with binary number
* **Some** – will understand the binary addition and multiplication.
 | * [Website for Binary Puzzles](http://www.binarypuzzle.com/)
* Video on [binary addition](http://www.youtube.com/watch?v=djI8YTs6Kdw)
* Binary multiplication explained [here](http://www.binarymath.info/multiplication-division.php)
* Scratch binary quiz Visualise binary counting [Here](http://scratch.mit.edu/projects/723745/)
 | * Assessment:

Binary addition [worksheet](https://docs.google.com/viewer?pid=explorer&srcid=0BxHHYOMUTUwNMkJCMFdKQjhUWDg&docid=9606511403d5733c20a1326d93391c11%7C0629b92b9899b1b4adf00c14d94c6794&chan=EgAAAENp2%2B83UZiFCIyh%2BLQQoSjQQzIlz8pV3DAspApYiGwg&a=v&rel=zip;z5;Assessment+Binary+Addition+Worksheet.pdf)0 + 0 = 01 + 0 = 10 + 1 = 11 + 1 = 101 + 1 + 1 = 11 | File size in number of bytes, kilobytes, Megabytes, gigabytes, addition, multiplication.  |
| Week6 | Introduction to algorithms using real life examples like making a coffee, crossing a road etc. know that algorithms are implemented using Pseudo code or flowcharts. Know the three basic constructs of programming (Sequence, Selection, Iteration). AT1

|  |
| --- |
|  |

 | * **All** – will be able to understand what is a algorithm and what is a sequence in an algorithm. Correct order of execution.
* **Most** – will be able to use the flowcharts to write simple algorithms
* **Some** – will progress to use the control structures . very few might be able to find the inefficiencies in algorithms.
 | * Intro to algorithm [powerpoint](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNOHVOUFdKSkJ3MG8)
* More on algorithms [here](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNOHVOUFdKSkJ3MG8)
* BBC bitesize [machine/ high level languages](http://www.bbc.co.uk/bitesize/intermediate2/computing/software_development/languages_and_environments/revision/1/)
 | **Home learning :** Give students an algorithm in Pseudo code - to reproduce in flowchart  | Pseudocode, flowcharts, algorithms, start/end, action, control, input, output , execute,sequence, selection, iteration.  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Learning objectives** – including literacy, numeracy and habits of mind (as appropriate) | **Differentiated teaching points/activities** including stretch and challenge | **Suggested resources** | **Suggested AfL activities** and formal assessment (when appropriate) | **keywords** |
| Week 7 | Understand and use the three programming constructs (sequence, selection and iteration) when using abstractions that model the state and behaviour of real-world problems and physical systems. Know two different types of languages in computing namely machine language and high level language AT1/AT2 | * **All** – understand the three programming constructs.
* **Most** – will be able to define and identify when to use each programming construct
* **Some** – will be able to use the constructs to write simple algorithm using either flow chart or a pseudo code of the given scenario..
 | * A ppt on P[rogramming constructs](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNOHVOUFdKSkJ3MG8)
* video on [sequence, selection and iteraction](http://www.youtube.com/watch?v=z4W1_bkMNO0)
* Definition [programming constructs](http://cnx.org/content/m19628/latest/)
 |  | algorithm, flowchart, pseudocode, sequence, selection,condition, iteration(loop), computer language, machine language, high level language, if, while , for, do while, if else, if elif etc |
| Week 8 | Intro to Python. 3.3 Using the IDLE. Print , input etc. Understand structure of a programming language. Learn how to run a programme using Python IDLE 3.3.3. Learn to save it using extension .py and be able to access it again. AT3  | * **All** – will be able to run a programme using python IDLE with/without teacher help. Understand that Python is one of many high level programming language.
* **Most** – will be able to print simple statements including their names
* **Some** – will be able to print statements with the user name as input from the user.
 | * [Introduction to Python](https://drive.google.com/?urp=http://www.google.com/accounts/Logout2?service%3Dwis&authuser=0&urp=http://www.google.com/accounts/Logout2?service%3Dwis#folders/0BxHHYOMUTUwNS0dVRlhKN1kyLTQ)
* Lesson plans with resources Python [here](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM) lesson 1.
 | * **Home learning:**

research online and make a list of various programming language or Download Python IDLE on their home computer <http://www.python.org/download/releases/3.3.0/> or Research different file extensions and list them .txt for text file .py for python , .jpeg etc | syntax, semantic, programming languageinput, print, run, execute, Python Shell, file extensions, CodeJAVA, C++, Csharp, Fortran, Pascal, Python, VBAlice, Greenfoot, Scratch, Kodu etc |
| Week 9 | know what is a variable and how it is different from constant. Learn to declare variables int, float, boolean, strings, arrays. Learn how to write comments within the programme. Know that comment is ignored by the compiler. Learn how to move the cursor to next line in Python Shell using /n.. AT3 | * **All** – know the various data types in python programming language.
* **Most** – declare and assign a value to a variable. What is a comment and how to write a comment in Python. use the /n for next line.

 understand variable and constant.* **Some** – understand why there is a need of various data types . When to use which datatypes.Investigate why indentation is important in Python.
 | * Python lesson2 on [data structures.](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM)
* String exercises [in Python](http://www.pythonschool.net/dayoneadditionalexercises/)
 | **Home learning:** Define the following words programme, code, variable, boolean.what is the difference between constant and variable? | syntax, semantics, variables, data types, statement, string, int, float, boolean, comment |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Learning objectives** – including literacy, numeracy and habits of mind (as appropriate) | **Differentiated teaching points/activities** including stretch and challenge | **Suggested resources** | **Suggested AfL activities** and formal assessment (when appropriate) | **Keywords** |
| Week 10 | Def a function. Be able to declare and call function in python.  write a simple programme which take user input and return something for example add the numbers and return the sum. understand the difference between function and procedure.Using built in functions by importing modules. Import time module AT3 | * **All** – understand how to define a function using the keyword def. How to call a function using ().
* **Most** – will be able to define and call functions with /without teacher support. Understand how to write a simple programme.Know difference between procedure and function..
* **Some** – will successfully write a programme in python which take input and return the sum. Use built in functions by importing classes. Import time , import maths etc
 | * Python [lesson on function](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM)
* Procedures in Python [here](https://drive.google.com/?urp=http://www.google.co.uk/url?sa%3Dt%26rct%3Dj%26q%3D%26esrc%3Ds%26s&pli=1&authuser=0#folders/0BxHHYOMUTUwNanlBVHQwRXVKWjA)
 | * **Home learning or Assessment**

 Write a program using either Pseudo Code or flowchart or python syntax which take input from a user (name and age ) and print the message that user will be x years old next year.Which data types you would need, what calculations you need to do, how you will tell the user of your programme how old he/she would be next year. | function, procedure, return value, declaring a function , calling a function, language keyword ‘def’,  import module, built in functions.  |
| Week 11 | Use the selection constructs of Python. If, Else, ,Elif . Understand why sequencing in not always useful and selection is required. Write a simple program using if, else and elif statements. AT3 | * **All** – will be able to use if statement in their code.
* **Most** – will be able to use if else statement in their programme.
* **Some** – will be able to use the ELIF nested if statement in their programme
 | * Python lesson 4,5,6 [If statements](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM)
 | * **Home learning or Assessment**

 Write your programme using python language. How can you improve it . Ask user gender and print he or she in your print message using if else statement.[Python Assessment](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM) - For all students | Language keywords IF, ELSE, ELIF, nested if  |
| Week 12 | Design and develop modular programs that use procedures or function . Import time and Maths module in their programme. Extension Intro to turtle and pygame.AT3 | * **All** – will be able to write a simple programme by importing a module with/without help teacher help.
* **Most** – Most will to write their own programme using the module time or maths.
* **Some** – will be able to work independently to design and develop modular programme which use functions to compute something and print answer for the user.
* **extension:** Intro to turtle and pygame. Use functions from module turtle or pygame.
 | * write a programme to work out area of a rectangle given length/ width or to find the area of a circle or write a programme to convert m to cm etc.

Some ideas on how to improve the code [here.](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNS0dVRlhKN1kyLTQ)Python Turtle [here](https://drive.google.com/?tab=wo&authuser=0#folders/0BxHHYOMUTUwNMTNrQXExVURkcHM) | * **End of term Assessment**

Write a programme which convert given cm to mm.peer marking- test each others programme. Give constructive feedback.(Soft towards people and hard towards content). Self reflection - What you have learnt and what you need to improve further. What do you want to learn next term | Import, class, module, built in, scope of a variable, function call etc |