



SVC ICT and Computing Learning Pathway -Year 7



| LP | Computer Science |
|-----|--|
| 8-9 | <p>Students will understand how bit patterns represent numbers and images.</p> <p>Students will understand & explain that iteration is the repetition of a process such as a loop e.g. smoke alarm (Algorithm/Flowchart).</p> <p>Students will have practical experience of a high-level textual language (Python Programming).</p> <p>Students will use a range of operators and expressions e.g. Boolean, and apply them in the context of program control e.g. Python Programming.</p> <p>Students will recognise and understand the function of the main internal parts of basic computer e.g. dismantling a PC.</p> <p>Students will understand the concepts behind the fetch-execute-cycle e.g. routine of instructions.</p> <p>Students will be able to suggest ways how search engines rank search results e.g. use of keywords, site structure, time spent on site, quality of links.</p> |
| 6-7 | <p>Students will analyse and represent symbolically a sequence of events e.g. create an algorithm using symbols (data flow diagram)</p> <p>Students will recognise different types of data e.g. text, number, real/integer, instructions and string.</p> <p>Students will understand the need for care and accuracy when programming e.g. errors, debugging.</p> <p>Students will give instructions involving selection and repetition e.g. loop, if, else.</p> <p>Students will ‘think through’ an algorithm and predict an output showing an awareness of inputs.</p> <p>Students will present data in a structured format suitable for processing.</p> <p>Students will recognise tasks completed by humans or computers e.g. data entry (register) can be automated or by humans, CAD/CAM – car production line.</p> <p>Students will design solutions by breaking down a problem and create a sub-solution for each of these parts.</p> <p>Students will design, write and debug structured (modular) programs using steps (procedures). E.g. lots of mini procedures which make up a program.</p> <p>Students will understand why and when computers are used e.g. work (production), social (communicating), efficiency & effectiveness of tasks.</p> <p>Students will understand how to effectively use search engines e.g. knowing how to use advanced search to narrow down specific data.</p> |
| 4-5 | <p>Students will recognise similarities between storyboards of everyday activities e.g. brushing teeth, cooking, walking to school.</p> <p>Students will present data in an orderly way e.g. storyboard, set of instructions i.e. method for recipe, making a sandwich or smoothie.</p> <p>Students will design a simple linear (non-branching) sequence of instructions to make things happen. E.g. Scratch, Small Basic, IF statement.</p> <p>Students will design simple algorithm flowchart to show solutions using repetition and two-way selection e.g. a flowchart with IF, then and else.</p> <p>Students will use algorithms to develop, improve instructions and create programs to achieve given goals.</p> <p>Students will recognise, state and assign variables.</p> <p>Students will use post-tested loop e.g. ‘until’, and a sequence of selection statements in programs, including and if, then and else statement e.g. Scratch – character to dance to music until it ends.</p> <p>Students will know that computers collect data from various input devices e.g. sensors – doors/lights and application software.</p> <p>Students will understand the difference between hardware and application software and their roles within a computer system e.g. labelling peripherals & components of computer system and what software to use for a particular purpose/task.</p> |

| LP | Computer Science |
|-----|---|
| 2-3 | <p>Students will draw a storyboard or other diagram of processes of activities e.g. algorithm showing wearing socks before shoes.</p> <p>Students will organise items in simple sets of data e.g. text, numeric, Boolean, binary.</p> <p>Students will plan and give direct commands to make things happen e.g. Big Track, Scratch.</p> <p>Students will solve simple problems using software e.g. Big Track, Scratch Formulas and Queries.</p> <p>Students will understand that algorithms are a set of instructions that are carried out on digital devices as programs e.g. smoke alarm.</p> <p>Students will design simple algorithms using loops and choices i.e. if statements e.g. brushing teeth.</p> <p>Students will be able to look at an algorithm, identify errors and correct them e.g. debugging, in algorithms.</p> <p>Students will identify and correct simple semantic errors (words, phrases, signs, and symbols).</p> <p>Students will use arithmetic operators (+ / * -) if statements, and loops, within programs e.g. Turtle/Small Basic creating a shape i.e. "square" & Big track.</p> <p>Students will recognise and can use a range of input and output devices e.g. digital camera, scanner, webcam, headphones, monitor, mouse, keyboard.</p> |
| 0-1 | <p>Students will talk about existing storyboards of everyday activities.</p> <p>Students will order a collection of pictures into the correct sequence.</p> <p>Students will recognise that many everyday devices respond to signals and instructions.</p> <p>Students will program a device or software to carry out instructions e.g. Scratch, Big Track</p> <p>Students will understand that an algorithm is a set of instructions that has to be carried in a specific order e.g. algorithm showing wearing socks before shoes.</p> <p>Students will understand that an algorithm uses different symbols for parts of a task e.g. start/end/process/decision</p> <p>Students will demonstrate programs does not rely on text e.g. Big Track moving forward & turning.</p> <p>Students will understand that programs are carried out by following exact instructions.</p> <p>Students will understand that computers have no brain and that we need to tell computers exactly what to do.</p> <p>Students will recognise that all software that is run on digital devices is programmed e.g. computers, iPad, mobile apps.</p> |