Computing Descriptors

	Grade 1	Grade 2	Grade 3	Grade 4	Grade 5	Grade 6	EP
Can develop computer based (ICT) resources…..	…when following detailed instructions, with assistance	…independently, when given detailed instructions	…independently when given outline instructions (i.e. using prior knowledge, researching any gaps in knowledge)	…independently based on own requirements / own design. Adds additional, relevant features to enhance products	…including consideration of good design (efficiency and maintainability)	Shows clear consideration of the needs of the user in designing the user interface. Can develop resources as part of effective team.	…including integration of differing technologies and researching techniques beyond those taught in the classroom
Programming	Can accurately copy simple examples of code using graphical programming languages and demonstrate they work.	Can copy simple examples of code and correct errors without intervention.	Can follow code examples in textural programming languages and explain what they do / how they work.	Can derive coded solutions to simple problems, when steered towards the appropriate technique. Can resolve simple syntax / logic errors	Can solve coding problems by independently choosing from available techniques. Can plan/design solutions on paper. Uses techniques to isolate and resolve more complex errors.	Structures code to aid readability and maintainability. Discusses the requirements behind a program in an abstract way (e.g. input requirements, output requirements, etc)	Researches and uses techniques beyond those taught in the classroom.
Review / testing		Can construct tests to ensure that code/products works	Can construct tests to identify any shortcomings of own products.	Consistently tests own work to identify any shortcomings. Can reflect on quality of work.	Can construct a comprehensive set of tests for desired functionality and references these in	Constructs comprehensive, formal test plans before developing code.	Can construct comprehensive tests for independently developed code,

					reflection/evaluation.		making use of test cases.
Computational skills		Can convert to/from binary.	Can accurately use taught information / calculation techniques in topics such as binary, Boolean logic, etc.	Can use calculation techniques including checking their accuracy.	Can also use hexadecimal. Can calculate data requirements for a given situation (E.g. graphics, file sizes, etc.) Can define/discuss generic techniques such as decomposition & abstraction	Can discuss <u>and apply</u> generic techniques such as decomposition & abstraction in their work.	