

Achievement Standard

Subject Reference Digital Technologies 1.44

Title Demonstrate understanding of basic concepts from computer science

Level 1 **Credits** 3 **Assessment** External

Subfield Technology

Domain Digital Technologies

Status Registered **Status date** 20 January 2011

Planned review date 31 December 2018 **Date version published** 20 November 2014

This achievement standard requires demonstrating understanding of basic concepts from computer science.

Achievement Criteria

Achievement	Achievement with Merit	Achievement with Excellence
<ul style="list-style-type: none"> Demonstrate understanding of basic concepts from computer science. 	<ul style="list-style-type: none"> Demonstrate in-depth understanding of basic concepts from computer science. 	<ul style="list-style-type: none"> Demonstrate comprehensive understanding of basic concepts from computer science.

Explanatory Notes

- 1 This achievement standard is derived from Level 6 of the Technology learning area in *The New Zealand Curriculum*, Learning Media, Ministry of Education, 2007; and is related to the material in the *Teaching and Learning Guide for Technology*, Ministry of Education at <http://seniorsecondary.tki.org.nz>.

Further information can be found at <http://www.technology.tki.org.nz/>.

Appropriate reference information is available in *Safety and Technology Education: A Guidance Manual for New Zealand Schools*, Ministry of Education at <http://technology.tki.org.nz/Curriculum-support/Safety-and-Technology-Education>, and the Health and Safety in Employment Act 1992.

- 2 *Demonstrate understanding of basic concepts from computer science* involves:
- describing the key characteristics and roles of algorithms, programs and informal instructions
 - describing an algorithm for a task, showing understanding of the kinds of steps that can be in an algorithm, and determining the cost of an algorithm for a problem of a particular size

- describing the role and characteristics of programming languages, including the different roles and characteristics of high level languages and low level (or machine) languages, and the function of a compiler
- describing the role of a user interface and factors that contribute to its usability.

Demonstrate in-depth understanding of basic concepts from computer science involves:

- explaining how algorithms are distinct from related concepts such as programs and informal instructions
- showing understanding of the way steps in an algorithm for a task can be combined in sequential, conditional, and iterative structures and determining the cost of an iterative algorithm for a problem of size n
- explaining how the characteristics of programming languages, including the different characteristics of high level and low level (or machine) languages, are important for their roles
- explaining the need for programs to translate between high and low level languages
- explaining how different factors of a user interface contribute to its usability.

Demonstrate comprehensive understanding of basic computer science concepts from computer science involves:

- comparing and contrasting the concepts of algorithms, programs, and informal instructions
- determining and comparing the costs of two different iterative algorithms for the same problem of size n
- comparing and contrasting high level and low level (or machine) languages, and explaining different ways in which programs in a high level programming language are translated into a machine language
- discussing how different factors of a user interface contribute to its usability by comparing and contrasting related interfaces.

- 3 The *basic concepts from computer science* are: the concept of an algorithm; the concept of a programming language; and the concept of a user interface and its usability.
- 4 An algorithm is a precise unambiguous specification of how to accomplish some computational task in a finite number of well-defined steps. An algorithm is distinct from a computer program. An algorithm has a cost (the number of steps it will perform) for a task. Different algorithms for the same task may have different costs.
- 5 A programming language is a precise, formal language for writing programs that can be run on a computer; it is distinct from pseudocode and natural language. There are different levels of programming languages; programs can be translated from high level to low level (or machine) languages.
- 6 A user interface is the part of a computer or electronic system that a human user interacts with to control the system. The usability of an interface is the key characteristic for evaluating an interface.

- 7 Assessment Specifications for this achievement standard can be accessed through the Technology Resources page found at <http://www.nzqa.govt.nz/qualifications-standards/qualifications/ncea/subjects/>.
-

Quality Assurance

- 1 Providers and Industry Training Organisations must have been granted consent to assess by NZQA before they can register credits from assessment against achievement standards.
- 2 Organisations with consent to assess and Industry Training Organisations assessing against achievement standards must engage with the moderation system that applies to those achievement standards.

Consent and Moderation Requirements (CMR) reference

0233