Achievement Standard

Subject Reference	Digital Technologies 3.7				
Title	Develop a complex computer program				
Level	3	**Credits**	4	**Assessment**	Internal
Subfield	Technology				
Domain	Digital Technologies				
Status	XX	**Status date**	XX		
Planned review date	XX	**Date version published**	XX		

This achievement standard involves developing a complex computer program.

Achievement Criteria

Achievement	Achievement with Merit	Achievement with Excellence
Develop a complex computer program.	Skillfully develop a complex computer program.	Accurately develop a complex computer program.

Explanatory Notes

1 This achievement standard is derived from 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/Technology-in-the-NZC/Safety-in-Technology-Education-revised-2017, and the Health and Safety at Work Act 2015.

This standard is also derived from *Te Marautanga o Aotearoa*. For details of *Te Marautanga o Aotearoa* outcomes to which this standard relates, see the Papa Whakaako for the relevant learning area.

2 *Develop a complex computer program* involves:
- writing code for a program that performs a specified task
- using complex techniques in a suitable programming language
- setting out the program code clearly and documenting the program with comments
- testing and debugging the program.

Skillfully develop a complex computer program involves:
- documenting the program with variable/module names and organised comments that describe code function
- following conventions for the chosen programming language
- testing and debugging the program in an organized way to ensure that it works on a sample of both expected and relevant boundary cases.

Accurately develop a complex computer program involves:
- ensuring that the program is a well-structured, logical solution to the task
- making the program flexible and robust
- comprehensively testing and debugging the program.

3 The programming language should be text based. The language chosen must support the required data types, control structures, complex techniques, and have good commenting facilities.

4 A *complex computer program* uses:
- variables storing at least two types of data (e.g. numeric, text, Boolean, object)
- sequence, selection and iteration control structures
- input from a user, sensors, or other external source, and produces output
- two or more complex techniques.

5 Examples of complex techniques include:
- programming or writing code for a graphical user interface (GUI)
- reading from or writing to files or other data storage
- object-oriented programming using class(es) defined by the student
- using a third party or non-core API, library or framework
- using complex data structures (e.g. hash tables, stacks, queues).

5 Example of ways of making a program flexible and robust include:
- using methods, functions, procedures, actions, conditions and control structures effectively
- checking input data for validity
- correctly handling expected, boundary and invalid values
- using constants, variables and derived values in place of literals.

6 Conditions of Assessment related to this achievement standard can be found at http://ncea.tki.org.nz/Resources-for-Internally-Assessed-Achievement-Standards.

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