

## **Level 1 Digital Technologies 91074 (1.44) Assessment Guide (Computer concepts)**

**Title** Demonstrate understanding of basic concepts from computer science

**Credits** 3

### **Teacher introduction**

Technology assessment guides have been produced to help teachers develop their own specific assessment guides. Examples of specific assessment guides, developed from the common assessment guide for each standard, have been produced as part of the external assessment resources for Level 1 Technology.

The specific assessment guides have been developed in a range of the contexts in which technology is practised. The specific assessment guides also show a variety of ways (i.e. case study, research practice) to produce external assessment material. The material in the candidate exemplars for each standard reflects the content and context of the specific assessment guides.

Teachers can adapt a common assessment guide and/or a specific assessment guide to suit their own context.

<b>Candidate introduction</b>
-------------------------------

## Assessment Activity 1

In this activity you will be given an algorithm or algorithms that solve a particular problem, and program(s) that run the algorithm(s).

### To demonstrate your understanding you will

1. Demonstrate the algorithm being used on a sample problem that you have selected. The role of the algorithm is to solve the problem, so demonstrating the algorithm establishes that you are aware of the role of the algorithm, and that you are aware of the key characteristics of an algorithm.
2. Describe the difference and similarities between the algorithm and the program(s) that run the algorithm, using your sample problem to illustrate this.
3. Using informal instructions, describe how the algorithm works. This will usually be a sentence, or a few sentences explaining the essence of the algorithm in a way that distinguishes it from other algorithms for the same problem.
4. Using the above as an example, explain how algorithms are distinct from programs and informal instructions, and compare and contrast these concepts.
5. Trial the algorithm(s) to measure their cost for some sample problems. The cost can be the time or number of key steps taken to run the program, or the number of steps needed to simulate the algorithm by hand; the trial could be done as a thought experiment, as a physical simulation, or by running a program. Do these trials for different sizes of input and characterise how the algorithm performs as the input gets large. Do this for more than one algorithm, and find out if there are differences in their relative performance.
6. List the kinds of instructions (steps) that you would need to have available if you had to tell someone else how to do the algorithm for you. What are the sequential, conditional, and iterative structures in the algorithm you have been given?

## Assessment Activity 2

In this activity you will be given a program written in a compiled language; you will also be given a compiler for the program. The program can be from activity 1, but this isn't required. For question 4 you will be given a second program in an interpreted language. The two languages should be readily available on your system.

- Interpreted languages include Scratch, Alice, AWK, JavaScript, and Ruby.
- Common languages that are usually compiled include Visual Basic, Visual C#, and C.
- Languages that use a mixture of the two approaches can be used but student will need to be aware that both compiling and interpreting is being used; such languages include Java, Python, Perl and Greenfoot.]

### To demonstrate your understanding you will

1. Describe what you have to do to get the program you were given to run, and what the role of a high-level and low-level language is in this process. What do you have to do to run the program again if it is changed?
2. Using the previous example, explain the characteristics of the programming language that the program was in, and the characteristics of the compiled (low level) program. Compare and contrast the different languages.
3. Explain the role of the compiler for your example.
4. Given a second program for the same task that is written in an interpreted languages, describe what you need to do to run that program, and explain how the characteristics of the two programming languages you have been given affect where you would use them.

## Assessment Activity 3

In this activity you will be given one or more programs or digital devices with an interface (they are likely to be a commercially available product, and if you review more than one device, they will have similar functionality but different interfaces).

### To demonstrate your understanding you will

1. Describe the context in which the device (or program) is used, and the role of its interface.
2. Describe the features of the interface that contribute to its usability in that context.
3. Explain the features of the interface that contribute to its usability in that context. What are the things that frustrate people when they use a digital device? What are the features that make an interface particularly easy to use or easy to learn?
4. Discuss the features of the interface that contribute to its usability in that context, comparing at least two of the interfaces you were given.

As you complete these activities you will produce a report

### **Candidate guidance for producing the report**

Your report will have three sections:

#### **Section 1: Algorithms**

- In this section, you will produce the section of the report that is related to Assessment activity one.

#### **Section 2: Programs**

- In this section, you will produce the section of the report that is related to Assessment activity two.

#### **Section 3: Interfaces**

- In this section, you will produce the section of the report that is related to Assessment activity three.

### **Essential documents**

The Achievement Standard governing the Common Assessment Guide can be found at:

<http://www.nzqa.govt.nz/nqfdocs/ncea-resource/achievements/2011/as91074.doc>

The Assessment Specifications for the Technology 91074 (1.44) Achievement Standard can be found at

<http://www.nzqa.govt.nz/nqfdocs/ncea-resource/specifications/2011/level1/technology.pdf>

## Definitions

The following definitions might be useful for you when you are writing your report:

- an **algorithm** is a precise unambiguous specification of how to accomplish a computational task in a finite number of well-defined steps. An algorithm is distinct from a computer program, and is often expressed in pseudo code. An algorithm has a cost (typically the number of steps it will perform or the time it takes) for a task. Different algorithms for the same task may have different costs.
- a **programming language** is a precise, formal language for writing programs that can be run on a computer; it is distinct from pseudo-code and natural language. There are different levels of programming languages; programs can be translated from high level to low level (or machine) languages. A programming language could be a drag-and-drop language, a specialised programming language, or a general purpose programming language.
- 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, and takes into account how humans function as well as how the computer functions.

## Further information

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

Appropriate reference information is available in:

- *Safety and Technology Education: A Guidance Manual for New Zealand Schools*, Learning Media, Ministry of Education, 1998
- the *Health and Safety in Employment Act 1992*
- *Technology Curriculum Support*, October 2007 (found at <http://techlink.org.nz/curriculum-support/papers/knowledge/tech-systems/index.htm>).

## Schedule

The Assessment Schedule for Technology 91074 (1.44) can be found at [91074-1.44-Assessment-Schedule.doc](#)

## Exemplars

Please read the exemplars below. You can model your work on these exemplars but you may not copy the material from the exemplar. Your report must be the product of your own efforts.

### Exemplars:

91074-1.44-Exemplar-1-Achievement.pdf

91074-1.44-Exemplar-2-Merit.pdf

91074-1.44-Exemplar-3-Excellence.pdf

Final grades will be decided using professional judgement based on a holistic examination of the evidence provided against the criteria in the achievement standard.