Australian Government

Department of Education, Employment and Workplace Relations

ICAB4229B Apply intermediate programming skills in another language

Release: 1

ICAB4229B Apply intermediate programming skills in another language

Modification History

Not Applicable

Unit Descriptor

Unit descriptor	This unit defines the competency required to undertake intermediate programming tasks using another programming language. The language may be an object-oriented language. No licensing, legislative, regulatory or certification requirements apply to this unit at the time of publication.

Application of the Unit

Application of the unit	

Licensing/Regulatory Information

Refer to Unit Descriptor

Pre-Requisites

Prerequisite units		
	ICAB4222B	Apply introductory programming skills in another language

Employability Skills Information

Employability skills	This unit contains employability skills.

Elements and Performance Criteria Pre-Content

Elements describe the essential outcomes of a unit of competency.	Performance criteria describe the performance needed to demonstrate achievement of the element. Where bold italicised text is used, further information is detailed in the required skills and knowledge section and the range statement. Assessment of performance is to be consistent with the evidence guide.

Elements and Performance Criteria

ELEMENT	PERFORMANCE CRITERIA
1. Code using user defined data structures	1.1. Design, define and use data structures that are an aggregate of other ***data types*** 1.2. Code using an array of user-defined ***data types*** 1.3. Use the facilities in the language to create, manipulate and destroy dynamic variables but limited to simple usage, such as an array of dynamic variables
2. Code using standard algorithms	2.1. Demonstrate use of a modular programming approach, including pass-by-reference parameter passing 2.2. Write code to create and manipulate a two-dimensional array 2.3. Create and maintain a sorted array and use language-provided facilities for sorting an array of ordered elements 2.4. Code a simple binary search technique for use with an array of sorted data 2.5. Code binary file handling solutions using random access algorithms
3. Debug code	3.1. Use standalone debugging tools or tools provided by ***integrated development environment*** to debug code 3.2. Use a debugger to trace code execution and examine variable contents to detect and correct errors
4. Document activities	4.1. Follow ***organisational guidelines*** for developing maintainable code and adhere to the provided ***coding standard*** when documenting activities 4.2. Apply internal ***documentation*** suitable for use by peers to all code created and utilise documentation tools available in the target ***language*** when documenting activities
5. Testing	5.1. Design and document tests 5.2. Undertake limited testing of produced code to ensure compliance with program specification 5.3. Capture and record test results
6. Create an application	6.1. Build an application that requires access to multiple source code files 6.2. Employ ***integrated development environment*** project maintenance facilities or make files to automate program building

ELEMENT	PERFORMANCE CRITERIA
	6.3. Develop a program specification solution when provided with a basic design document 6.4. Design the algorithm, and document, construct and test applications in response to a problem description using the ***language***

Required Skills and Knowledge

REQUIRED SKILLS AND KNOWLEDGE
This section describes the skills and knowledge required for this unit.
Required skills
• Reading and interpreting program specifications • Translating requirements from problem space to machine space • Integrated development environment usage • Program coding techniques • Internal (code) documentation techniques • Debugging techniques • Testing techniques • Documentation techniques
Required knowledge
• Medium application development • User-defined data structures • Standard array and file handling algorithms • Dynamic variables

Evidence Guide

EVIDENCE GUIDE	
The evidence guide provides advice on assessment and must be read in conjunction with the performance criteria, required skills and knowledge, range statement and the Assessment Guidelines for the Training Package.	
Overview of assessment	
Critical aspects for assessment and evidence required to demonstrate competency in this unit	Evidence of the following is essential: • Assessment must confirm that application programs are designed and built from a provided problem scenario and program specification. • Design and code documentation must be generated. Testing must be used to confirm that created application meets original specification and solves original problem. • Assessment must confirm competency in all areas of the software development cycle. To demonstrate competency in this unit the person will require access to: • Software development environment • Technical requirements
Context of and specific resources for assessment	Programming languages form the underpinning software infrastructure of all computer data processing. The breadth, depth and complexity of knowledge and skills in this competency would cover a broad range of varied activities or application in a wider variety of contexts most of which are complex and non-routine. Leadership and guidance would be involved when organising activities of self and others as well as contributing to technical solutions of a non-routine or contingency nature. Assessment must ensure: • Performance of a broad range of skilled applications including the requirement to evaluate and analyse current practices, develop new criteria and procedures for performing current practices and provision of some leadership and guidance to others in the application and planning of the skills would be characteristic.

EVIDENCE GUIDE	
	• Applications may involve responsibility for, and limited organisation of, others. The stages of the development methodology should be followed within the scope of a project or scenario, and the relevant supporting documentation produced.
Method of assessment	The purpose of this unit is to define the standard of performance to be achieved in the workplace. In undertaking training and assessment activities related to this unit, consideration should be given to the implementation of appropriate diversity and accessibility practices in order to accommodate people who may have special needs. Additional guidance on these and related matters is provided in ICA05 Section 1. • Competency in this unit should be assessed using summative assessment to ensure consistency of performance in a range of contexts. This unit can be assessed either in the workplace or in a simulated environment. However, simulated activities must closely reflect the workplace to enable full demonstration of competency. • Assessment will usually include observation of real or simulated work processes and procedures and/or performance in a project context as well as questioning on underpinning knowledge and skills. The questioning of team members, supervisors, subordinates, peers and clients where appropriate may provide valuable input to the assessment process. The interdependence of units for assessment purposes may vary with the particular project or scenario.
Guidance information for assessment	Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended. An individual demonstrating this competency would be able to: • Demonstrate understanding of a broad knowledge

EVIDENCE GUIDE	
	base incorporating some theoretical concepts • Apply solutions to a defined range of unpredictable problems • Identify and apply skill and knowledge areas to a wide variety of contexts, with depth in some areas • Identify, analyse and evaluate information from a variety of sources • Take responsibility for own outputs in relation to specified quality standards • Take limited responsibility for the quantity and quality of the output of others • Maintain knowledge of industry products and services Additionally, an individual demonstrating this competency would be able to: • Understand a limited range of development methodologies and their application • Demonstrate limited theoretical knowledge of language development • Apply a programming methodology to a project or scenario • Produce documentation required for the chosen methodology

Range Statement

RANGE STATEMENT	
The range statement relates to the unit of competency as a whole. It allows for different work environments and situations that may affect performance. Bold italicised wording, if used in the performance criteria, is detailed below. Essential operating conditions that may be present with training and assessment (depending on the work situation, needs of the candidate, accessibility of the item, and local industry and regional contexts) may also be included.	
Organisational guidelines may include but are not limited to:	• personal use of emails and internet access • content of emails • downloading information and accessing particular websites

RANGE STATEMENT	
	• opening mail with attachments • virus risk • dispute resolution • document procedures and templates • communication methods • financial control mechanisms
Documentation may follow:	• ISO/IEC/AS standards • audit trails • naming standards • version control • project management templates • report writing
Data types may include but are not limited to:	• language-provided data types
Integrated development environment may include but is not limited to:	• Borland C++ • Visual C++ • Visual Studio suite • Eclipse • Code Warrior
Coding standards may include:	• ANSI C coding standard • GNU coding standard
Language may include but is not limited to:	• C • VB • Java • C++ • Small Talk • VB.net

Unit Sector(s)

Unit sector	Build

Co-requisite units

Co-requisite units		

Competency field

Competency field	