Information Technology

Pathway: Programming and Software Development

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.01	Programming and Software Engineering Pathway	#4: Fundamentals of Programming and Software Development	Identify and analyze customer software needs and requirements.	Gather data to identify customer requirements.	Gather information using interviewing strategies.
					Identify input and output requirements.
					Identify system processing requirements.
					Clarify specifications using questioning techniques.
					Identify hardware, networking, and software system functional requirements.
					Demonstrate knowledge of nonfunctional requirements (e.g., security, integrity response time, reliability, support, and documentation).
				Conduct needs analysis.	Gather information on problems from users.
					Perform workflow analysis to determine user needs.
					Analyze existing procedures.
					Define business problem to be solved by the application.
				Develop software requirements specification.	Demonstrate knowledge of the use, structure, and contents of a requirements specification document.
					Define system and software requirements.
					Develop informal specifications.
					Develop formal specification.
					Resolve conflicting requirements.
					Review and verify specification with customer.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
			(CONTINUED)	Analyze requirements/specifications using current approaches.	Demonstrate knowledge of how to use software methodologies to analyze a real-world problem.
					Identify constraints.
					Demonstrate knowledge of modeling and analyzing functional requirements (e.g., dataflow diagrams, process specifications, and a data dictionary).
					Demonstrate knowledge of modeling and analyzing data requirements (e.g., Jackson diagrams, entity relationship diagrams, and relations).
				Use available reference tools as appropriate.	Access needed information using company and manufacturers' references (e.g., procedural manuals, documentation, standards, work flowcharts).
ITPD01.02	Programming and Software Engineering Pathway	#4: Fundamentals of Programming and Software Development	Produce IT-based strategies and project plan to solve the specific problem.	Define scope of work for the programming project.	Define scope of work to meet customer needs.
					Demonstrate knowledge of the key functions and subsystems of the software product.
					Demonstrate knowledge of software development process and issues.
					Demonstrate knowledge of the system life-cycle approach.
				Demonstrate knowledge and skills of working in a software development team.	Identify resources and risks.
					Demonstrate knowledge of cross-functional team structures and team members' roles.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
			(CONTINUED)	Design project plan.	Demonstrate knowledge of project budgeting, scheduling, and control issues related to software development.
					Demonstrate knowledge of software development methodology.
					Develop implementation plan.
ITPD01.03	Programming and Software Engineering Pathway	#4: Fundamentals of Programming and Software Development	Define and analyze system and software requirements.	Identify new IT technologies and assess their potential importance and impact in the future.	Identify new technologies relevant to information technology.
					Assess the importance of new technologies to future developments.
					Identify system processing requirements.
					Identify data communication trends and major current issues.
				Explain new and emerging classes of software.	Identify new and emerging classes of software.
					Determine compatibility of hardware and software.
				Understand elements and types of information processing.	Identify the elements of the information processing cycle (i.e., input, process, output, and storage).
				Explain measurement techniques for increased productivity due to information systems implementation.	Identify metrics for measurements.
					Measure increases in productivity realized by the implementation of information systems.
					Identify new and emerging drivers and inhibitors of information technology change.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.04	Programming and Software Engineering Pathway	#5: Software Development Tools	Demonstrate the effective use of tools for software development.	Apply tools for developing software applications.	Demonstrate knowledge of software development environment.
					Use prototyping techniques.
					Use appropriate configuration management tools.
					Use appropriate issues tracking tools.
					Demonstrate knowledge of reuse and components.
				Apply language-specific programming tools/techniques.	Develop programs using appropriate language.
					Use appropriate development environment for the selected language (e.g., compilers, debuggers, test generation, static analyzer, etc.).
					Use user interface development tools.
				Demonstrate knowledge of how to use computer-aided software engineering (CASE) tools.	Use appropriate requirement analysis tools.
					Use appropriate modeling and analysis tools.
					Use requirement tracking tools.
					Demonstrate knowledge of software reuse, design pattern, and components.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.05	Programming and Software Engineering Pathway	#6: Introduction to Software Design	Demonstrate knowledge of the software development process.	Demonstrate knowledge of software development methodology.	Demonstrate knowledge of the information system life cycle.
					Demonstrate knowledge of system analysis issues related to design, testing, implementation, and maintenance.
					Record and analyze process.
					Identify the use of program design tools in a software development process.
					Identify roles on team members/customers in the software development process.
					Identify current information life cycle models.
ITPD01.06	Programming and Software Engineering Pathway	#6: Introduction to Software Design	Design a software application.	Create design specification for a computer application.	Demonstrate knowledge of the principles of program design (e.g., structured, object-oriented, event-driven) to analyze real-world problems.
					Perform a logical design.
					Document design specification according to defined procedures.
					Design system input, output, processing, and interfaces.
					Review system design with management and users.
					Use current computer-aided software engineering (CASE) tools.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
			(CONTINUED)	Demonstrate comprehension of the trade-off involved in design choices.	Demonstrate knowledge of the characteristics and the uses of processing (e.g., batch, interactive, event-driven, object-oriented).
					Identify basic concepts of algorithm and data structure development.
					Demonstrate knowledge of different data types (e.g., numeric, alphanumeric, tables, etc.).
					Identify constraints.
					Demonstrate knowledge of nonfunctional requirements (e.g., security, integrity, response time, cost, dependability, migration, etc.).
					Demonstrate knowledge of modular design concepts.
					Demonstrate knowledge of the features, functions, and architectures of client/server computing.
				Apply the principles of effective information management, information organization, and information-retrieval skills.	Demonstrate knowledge of database management concepts.
					Define the objectives of a client/server application.
					Implement online transition processing.
					Design static and dynamic online processing systems.
					Employ interface techniques.
				Demonstrate knowledge of computing/networking hardware and software architecture.	Demonstrate knowledge of how data is stored, organized, managed and retrieved in various computer and network architecture.
					Demonstrate knowledge of software architectures for different classes of applications.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.07	Programming and Software Engineering Pathway	#7: Program and Software Development	Produce (code) a computer application.	Demonstrate knowledge of programming language concepts.	Demonstrate knowledge of the hardware-software connection.
					Demonstrate knowledge of the concepts of data and procedural representation.
					Demonstrate knowledge of the basic principles for analyzing a programming language.
					Demonstrate knowledge of the basics of structured, object-oriented, and event-driven programming.
					Demonstrate knowledge of how a programming language can support multitasking and exception-handling.
				Demonstrate proficiency in developing an application using an appropriate programming language.	Demonstrate knowledge of current key programming languages and the environment in which they are used (e.g., C, C++, Visual Basic, Java, RPG, COBOL, Assembler).
					Translate data structure and program design into code in an appropriate language.
					Demonstrate knowledge of key constructs and commands specific to a language.
				Understand the range of languages used in software development.	Demonstrate knowledge of the range of languages used in software development.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
			(CONTINUED)	Demonstrate knowledge of program development methodology.	Demonstrate knowledge of how to resolve program implementation issues (e.g., debugging, documentation, auditing).
					Demonstrate knowledge of software development issues (e.g., correctness, reliability, and productivity).
					Demonstrate knowledge of code analysis issues related to design, testing, implementation, and maintenance.
					Demonstrate knowledge of how to design and implement programs in a top-down manner.
					Demonstrate knowledge of how to translate algorithmic and modular design to develop a program.
					Demonstrate knowledge of structured/modular programming.
					Demonstrate knowledge of how programming control structures are used to verify correctness.
					Use code development tools (debugger, integrated development environments).
				Demonstrate knowledge of basic software systems implementation.	Use appropriate programming language.
					Analyze and prepare logic using program flowchart.
					Review design (e.g., peer and/or user walkthrough).
					Compile and debug code.
					Prepare code documentation.
					Prepare unit testing plan.
					Conduct unit testing and bug fixes.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
			(CONTINUED)	Develop software requirements/specifications.	Access needed information using company and manufacturers' references (e.g., procedural manuals, documentation, standards, word flowcharts).
					Divide design specifications into logical process blocks.
					Identify parameters.
					Follow specifications or drawings.
					Record process (e.g., using flowchart, step-by-step narrative).
					Record data.
				Resolve problems with integration if they occur.	Troubleshoot unexpected results.
					Fix code.
ITPD01.08	Programming and Software Engineering Pathway	#8: Testing and Quality Assurance	Demonstrate knowledge of software testing.	Develop test plan.	Access needed information using appropriate reference materials.
					Define test procedures.
					Analyze requirement and design specifications.
					Development test cases using requirements and design specification.
				Perform testing and validation.	Perform integration testing.
					Perform regression testing.
					Help with user-acceptance test.
					Validate user documentation.
					Document discovered errors.
				Document test results.	Document errors discovered.
					Perform defect tracking.
				Develop software testing audit trails.	Record error correction procedures and actions.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.09	Programming and Software Engineering Pathway	#8: Testing and Quality Assurance	Perform quality assurance tasks to produce quality products.	Demonstrate knowledge of software quality assurance.	Demonstrate knowledge of Software QA process.
					Demonstrate knowledge of the standards/requirements for Software QA.
					Develop team relationships to support software quality assurance tasks.
				Perform software quality assurance tasks to produce a quality software product.	Identify standards and issues related to I/O programming and design of I/O interfaces.
					Use customer satisfaction in determining product characteristics (e.g., cost, user-friendliness).
					Recognize the relationship between dependability, functionality, ease of use, etc.
					Conduct code walkthrough and/or inspection.
					Follow established procedures for testing, identifying problems, and tracking resolutions.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved

Code	Topic	Course	Knowledge and Skill	Performance Element	Measurement Criteria
ITPD01.10	Programming and Software Engineering Pathway	#9: Maintenance and Support of Programming and Software Development	Perform maintenance and customer support functions.	Analyze software technical support needs.	Identify maintenance and support requirements.
					Apply information and data analysis techniques.
					Define scope of work to meet customer support needs.
				Perform customer service.	Access needed information using appropriate reference materials.
					Provide help to first line user-support personnel to answer user questions.
					Provide troubleshooting for software.
					Perform system-tuning function.
					Diagnose problems within system.
					Perform technical functions required by customer/user.
					Communicate and document technical support provided.
				Perform software maintenance activities.	Follow organizational procedures in communicating and documenting maintenance tasks.
					Identify and analyze problem.
					Analyze and propose solutions.
					Implement solutions in code and documentation.
					Release software and documentation updates according to procedures.

Copyright 2007, States' Career Clusters Initiative. All Rights Reserved