Computer Programming 2 Python

Exam Information	Description
Exam number **827** **Items** **40** **National Career Cluster** **Information Technology** **Certification available** **Yes**	The Computer Programming 2 Python industry certification exam assesses advanced programming concepts in Python. Learners demonstrate their ability to create more powerful programs using Python by applying advanced techniques and concepts.

Exam Blueprint and Performance Skills

Standard	Percentage of exam score
1. List Structures and Strings	
2. Sequential Files	
3. User Defined Functions	
4. Object-Oriented Programming	
5. Code Comprehension and Debugging	
6. Team Programming	
7. Computer and Information Ethics	
8. Career Opportunities and History	

*Includes performance skill evaluation

Standard 1 - List Structures and Strings

Candidates will demonstrate static (array), dynamic (vector, Arraylist, etc.) list structures, and strings.

Objective 1 Demonstrate the ability to use static arrays/lists in programs.

1. Declare and initialize arrays/lists of all applicable types
2. Perform data input to and output from arrays/lists
3. Perform operations on arrays/lists including sort arrays
4. Iterate through the structure (i.e., for-each, enhanced for, or iterators)

Objective 2 Demonstrate the ability to use dynamic arrays/lists (i.e. vectors, Arraylist, or generic lists).

1. Declare and initialize a dynamic array/list
2. Add and remove items from the array/list
3. Output data from arrays/lists
4. Perform operations on arrays/lists
5. Iterate through the structure (i.e. for-each, enhanced for, or iterators)
6. Use a loop to iterate through the structure

Objective 3 Demonstrate the ability to use strings in programs.

1. Compare string values
2. Find the length of a string
3. Copy part or all of string values into other strings
4. Concatenate string values
5. Locate substring positions
6. Insert strings into other strings

Optional: Standard performance evaluations on the final page of this document

Standard 2 - Sequential Files

Candidates will properly use sequential files.

Objective 1 Demonstrate the ability to use sequential files in programs.

1. Create and initialize sequential files
2. Store data to sequential files
3. Retrieve data from sequential files
4. Update sequential files

Objective 2 Demonstrate how to use an API in program development.

1. If candidates do not have access to use APIs directly, the teacher may provide a demonstration instead.

Standard 3 - User-Defined Functions

Create user defined functions/methods/procedures using top-down design and functional decomposition.

Objective 1 Candidates will understand and properly apply scope.

1. Understand that variables and functions/methods/procedures have scope, which influences where they can be declared and accessed
2. Declare and access local variables in a program
3. Declare and access global variables in a program

Objective 2 Candidates will understand and implement function/methods/procedures inputs and outputs.

1. Understand the correlation between arguments (inputs) and parameters (variables)
2. Understand that functions/methods/procedures may or may not require arguments
3. Understand that functions/methods/procedures may or may not return values
4. Define functions/methods/procedures:
    a. with parameters
    b. without parameters
    c. with return values
    d. without return values
    e. default parameters

Objective 3 Candidates will understand and implement functional decomposition. (Breaking a program down into one or more functions/methods/procedures.)

1. Identify repetitive or redundant code in an application
2. Understand the role abstraction plays in computer programming
3. Demonstrate how to abstract multiple steps into a function/method/procedure
4. Identify the characteristics of a well-defined function/method/procedure
    a. Examples: shorter code, efficiency, reduced memory consumption, high reliability, readability, abstraction

Standard 4 - Object-Oriented Programming

Candidates will properly demonstrate object-oriented programming techniques.

Objective 1 Demonstrate the ability to use built-in classes.

1. Instantiate objects
2. Use object data members (i.e., Java's arr. length)
3. Use object member functions (methods)

Objective 2 Demonstrate the ability to create user-defined classes.

1. Create and use data members (instance variables)
2. Create a constructor to initialize the data members
3. Create and use member functions (methods)

Standard 5 - Code Comprehension and Debugging

Candidates will properly demonstrate code comprehension and debugging techniques.

Objective 1 Demonstrate the ability to comprehend code outcomes.

1. Tracing - Cognitively following the passes of a loop, nested function calls, change in value of global and local scoped variables, etc.
2. Debugging - Utilizing 3rd party tools (IDE's) to step through a program and troubleshoot

3. Testing - Validating the outputs of a program and testing its robustness. (i.e., boundary conditions, invalid inputs, unexpected scenarios, incorrect results, etc.)

Standard 6 - Team Programming

Candidates will apply appropriate programming skill as an effective member of a team demonstrating the ability to collaborate with others.

Objective 1 Demonstrate the ability to apply knowledge to a programming project.

1. Formalize specifications
2. Choose proper input parameters
3. Choose appropriate data structures and processing
4. Design appropriate output
5. Use appropriate test data
6. Write good documentation

Objective 2 Demonstrate the ability to use teamwork and collaboration in a programming project.

1. Divide a project among programmers
2. Present work to a group
3. Coordinate work with others in the group
4. Complete assigned work according to predetermined deadlines
5. Participate in a peer performance evaluation
6. Demonstrate professionalism in team relationships, communication, timeliness, and attitude

Standard 7 - Computer and Information Ethics

Candidates will demonstrate knowledge of current ethical issues dealing with computers and information in a global society.

Objective 1 Demonstrate knowledge of the social and ethical consequences of computers.

1. Explain the ethical reasons for creating reliable and robust software

2. Explain the impact software can have on society (i.e., privacy, piracy, copyright laws, ease of use, ete.)
3. Show how security concerns can be addressed in an application (i.e., biometrics, passwords, information hiding, etc.)
4. Describe how computer-controlled automation affects a workplace and society
5. Give examples of ways to protect information on computer systems (attacks, viruses, malware, etc.)

Optional: Standard performance evaluations on the final page of this document

Standard 8 - Career Opportunities and History

Candidates will be aware of career opportunities in the Computer Programming/Software Engineering industry and of its history.

Objective 1 Investigate career opportunities, trends, and requirements related to computer programming/software engineering careers.

1. Identify the members of a computer programming/software engineering team: team leader, analyst, senior developer, junior developer, and client/subject matter expert
2. Describe work performed by each member of the computer programming/software engineering team
3. Investigate trends and traits associated with computer programming/software engineering careers
4. (creativity, technical, leadership, collaborative, problem solving, design, etc.)
5. Discuss related career pathways

Performance Skills Evaluation

Exam name: Computer Programming 2 Python

Candidate Name: _____

Performance assessments may be completed and evaluated any time before the certification is issued. The following performance skills can be used in connection with the associated standards and exam. The evaluator will function as the subject matter expert to determine if the candidate meets the skills requirements with a Yes/No for passing.

Standard 1 – List Structures and Strings • Candidates demonstrate mastery of static and dynamic arrays, lists, and strings in projects.	**Pass:**
Standard 2 – Sequential Files • Demonstrate sequential file access utilizing reading and writing operations.	**Pass:**
Standard 3 – User-Defined Functions • Create several user defined functions/methods/procedures with and without inputs and/or return values.	**Pass:**
Standard 4 – Object-Oriented Programming • Properly employ object-oriented programming techniques.	**Pass:**
Standard 5 – Code Comprehension and Debugging • Demonstrate code comprehension and debugging techniques by tracing, debugging, and testing programs.	**Pass:**
Standard 6 – Team Programming • Apply appropriate programming skills as an effective member of a team.	**Pass:**
Standard 7 – Computer and Information Ethics • Demonstrate knowledge of current ethical issues dealing with computers and information in society.	**Pass:**
Standard 8 – Career Opportunities and History • Develop awareness of career opportunities in the computer programming/software engineering industry and of its history	**Pass:**

Evaluator Name: _____

Date: _____