

National Occupational Standards



SIC - Coding and Programming (Essentials of Coding and Programming)

Unit Code: ELE/N0801

Version: 1.0

NSQF Level: 4

Electronics Sector Skills Council of India || 155, 2nd Floor, ESC House Okhla Industrial Area-Phase 3
New Delhi- 110020 || email:ceo@essc-india.org



National Occupational Standards

Description

The SIC - Coding and Programming is an upskilling course that offers an individual knowledge for developing applications and platforms in high level language in order to build a robust hack free system. They will be responsible for evaluating the technical performance of algorithmic models on the system on which it is being deployed. They will be responsible for developing, designing, building, testing and deploying programming solutions.

Scope

The scope covers the following :

- The scope covers the following:
- IoT Platform Sequential Planning Programming
- List, Tuple, Dictionary and Sequential Data Types
- Algorithm and Data Structures
- Python and Pandas for Data Processing
- Data Analysis and Visualization Mini Projects

Elements and Performance Criteria

Module 1: Sequential Planning Programming

To be competent, the user/individual on the job must be able to:

- PC1.** Intro to sequential programming and its significance in software development, demonstrate proficiency in writing sequential code using Python
- PC2.** Identify common programming errors in sequential code and debug them effectively, apply sequential programming concepts to solve simple computational problems
- PC3.** Evaluate and refine the project plan based on feedback and changing requirements, Document the planning process and project requirements effectively for future reference.
- PC4.** Define numeric data types, variables and evaluate the expressions and basic arithmetic and logical operations using Python.
- PC5.** Understand, Debug and troubleshoot conditional and decision-making statements to ensure correct program behavior under different scenarios, Extend & Implement conditional logic with if and else statements to handle multi-directional decision making.
- PC6.** Define and implement looping structures and their significance in iterative processing of data in programs, Debug and optimize looping constructs and conditional evaluation to achieve desired program behavior and efficiency.
- PC7.** Apply looping techniques to solve real-world programming challenges effectively.

List, Tuple, Dictionary and Sequential Data Types

To be competent, the user/individual on the job must be able to:

- PC8.** Define lists and tuples as sequence data types and Perform common operations on lists and tuples in Python.
- PC9.** Apply list comprehensions and tuple unpacking techniques to manipulate data structures efficiently.
- PC10.** Describe dictionaries as key-value pairs and their role in efficient data retrieval, manipulate dictionary data using built-in methods and key-based operations in Python.

National Occupational Standards

PC11. Implement and evaluate dictionary comprehensions, dictionary unpacking and sequence operations to iterate over dictionary elements effectively for different programming tasks.

Algorithm and Data Structures

To be competent, the user/individual on the job must be able to:

- PC12.** Explain the concepts of stack and queue and perform using stack and queue operations using Python efficiently.
- PC13.** Apply stacks and queues to solve problems requiring Last-In-First-Out (LIFO) and First-In-First-Out (FIFO) data structures, Analyze the time and space complexity of stack and queue operations for algorithmic efficiency.
- PC14.** Understand and implement the principles of sequential search and binary search algorithms, Compare and analyze the performance characteristics of sequential and binary search algorithms, considering factors such as time complexity and space complexity.
- PC15.** Explain the fundamental principles behind sorting algorithms such as bubble sort, selection sort, and insertion sort, implement merge sort and quick sort algorithms for efficient sorting of data, compare and analyze the performance of sorting algorithms under various input conditions and data distributions.
- PC16.** Understand problem-solving paradigms such as greedy approach, divide-and-conquer, dynamic programming, and backtracking, Apply greedy algorithms to solve optimization problems by making locally optimal choices.
- PC17.** Implement divide-and-conquer algorithms to solve complex problems by breaking them into smaller subproblems, Utilize dynamic programming techniques to solve problems with overlapping subproblems and optimal substructure.

Python and Pandas for Data Processing

To be competent, the user/individual on the job must be able to:

- PC18.** Explore built-in Python modules and third-party libraries for data processing and analysis, Import and utilize Python modules such as os, sys, and math for system operations, file handling, and mathematical computations.
- PC19.** Install and import external libraries like NumPy and SciPy for scientific computing and advanced mathematical operations, Evaluate the documentation and community support of Python modules to ensure their reliability and compatibility with project requirements.
- PC20.** Understand the Pandas library and its significance in data manipulation and analysis, create Pandas series and Perform data cleaning and tidying operations using Pandas functions like dropna(), fillna(), and replace().

Data Analysis and Visualization Mini Projects

To be competent, the user/individual on the job must be able to:

- PC21.** Apply tidy data principles to organize and structure data for efficient analysis and visualization, Analyze and visualize time series data using Pandas.
- PC22.** Apply data analysis techniques to real-world datasets related to financial markets, healthcare, or environmental trends, Perform and visualize statistical and data analysis and hypothesis testing to validate research findings and business hypotheses.
- PC23.** Present mini-project findings through written reports and oral presentations to stakeholders and peers.

Knowledge and Understanding (KU)

The individual on the job needs to know and understand:

National Occupational Standards

- KU1.** Understanding the purpose and use of different programming languages.
- KU2.** Familiarity with syntax and semantics of at least one programming language (e.g., Python, Java, C plus plus).
- KU3.** Comprehending the use of variables, data types, and operators.
- KU4.** Understanding how to perform basic input and output operations.
- KU5.** Knowledge of conditional statements (if, else, switch).
- KU6.** Knowledge of conditional statements (if, else, switch).
- KU7.** Understanding the concepts of classes and objects.
- KU8.** Knowledge of OOP principles: inheritance, polymorphism, encapsulation, and abstraction.
- KU9.** Ability to write, test, and debug code in one or more programming languages.
- KU10.** Proficiency in using basic programming constructs and data structures.
- KU11.** Familiarity with Integrated Development Environments (IDEs).
- KU12.** Understanding the basics of version control systems (e.g., Git) for code management.
- KU13.** Knowledge of the principles of computational thinking and how they apply to programming.
- KU14.** Understanding the impact of efficient code on software performance.

Generic Skills (GS)

User/individual on the job needs to know how to:

- GS1.** Ability to identify, analyze, and solve complex problems using logical and structured approaches.
- GS2.** Proficiency in breaking down problems into smaller, more manageable components.
- GS3.** Capacity to think critically and evaluate different solutions to determine the most effective approach.
- GS4.** Meticulousness in writing and reviewing code to ensure accuracy and avoid errors.
- GS5.** Ability to apply logical reasoning to develop algorithms and flowcharts.
- GS6.** Effective management of time to balance multiple assignments, projects, and deadlines.
- GS7.** Clear and concise communication skills to explain coding concepts and solutions to others.
- GS8.** Clear and concise communication skills to explain coding concepts and solutions to others.
- GS9.** Understanding the importance of writing secure, ethical, and responsible code.
- GS10.** Ability to keep track of progress and document code changes effectively.

National Occupational Standards

Assessment Criteria

Assessment Criteria for Outcomes	Theory Marks	Practical Marks	Project Marks	Viva Marks
<i>Module 1: Sequential Planning Programming</i>	21	-	-	-
PC1. Intro to sequential programming and its significance in software development, demonstrate proficiency in writing sequential code using Python	3	-	-	-
PC2. Identify common programming errors in sequential code and debug them effectively, apply sequential programming concepts to solve simple computational problems	3	-	-	-
PC3. Evaluate and refine the project plan based on feedback and changing requirements, Document the planning process and project requirements effectively for future reference.	3	-	-	-
PC4. Define numeric data types, variables and evaluate the expressions and basic arithmetic and logical operations using Python.	3	-	-	-
PC5. Understand, Debug and troubleshoot conditional and decision-making statements to ensure correct program behavior under different scenarios, Extend & Implement conditional logic with if and else statements to handle multi-directional decision making.	3	-	-	-
PC6. Define and implement looping structures and their significance in iterative processing of data in programs, Debug and optimize looping constructs and conditional evaluation to achieve desired program behavior and efficiency.	3	-	-	-
PC7. Apply looping techniques to solve real-world programming challenges effectively.	3	-	-	-
<i>List, Tuple, Dictionary and Sequential Data Types</i>	16	-	-	-
PC8. Define lists and tuples as sequence data types and Perform common operations on lists and tuples in Python.	4	-	-	-
PC9. Apply list comprehensions and tuple unpacking techniques to manipulate data structures efficiently.	4	-	-	-

National Occupational Standards

Assessment Criteria for Outcomes	Theory Marks	Practical Marks	Project Marks	Viva Marks
PC10. Describe dictionaries as key-value pairs and their role in efficient data retrieval, manipulate dictionary data using built-in methods and key-based operations in Python.	4	-	-	-
PC11. Implement and evaluate dictionary comprehensions, dictionary unpacking and sequence operations to iterate over dictionary elements effectively for different programming tasks.	4	-	-	-
<i>Algorithm and Data Structures</i>	24	-	-	-
PC12. Explain the concepts of stack and queue and perform using stack and queue operations using Python efficiently.	4	-	-	-
PC13. Apply stacks and queues to solve problems requiring Last-In-First-Out (LIFO) and First-In-First-Out (FIFO) data structures, Analyze the time and space complexity of stack and queue operations for algorithmic efficiency.	4	-	-	-
PC14. Understand and implement the principles of sequential search and binary search algorithms, Compare and analyze the performance characteristics of sequential and binary search algorithms, considering factors such as time complexity and space complexity.	4	-	-	-
PC15. Explain the fundamental principles behind sorting algorithms such as bubble sort, selection sort, and insertion sort, implement merge sort and quick sort algorithms for efficient sorting of data, compare and analyze the performance of sorting algorithms under various input conditions and data distributions.	4	-	-	-
PC16. Understand problem-solving paradigms such as greedy approach, divide-and-conquer, dynamic programming, and backtracking, Apply greedy algorithms to solve optimization problems by making locally optimal choices.	4	-	-	-
PC17. Implement divide-and-conquer algorithms to solve complex problems by breaking them into smaller subproblems, Utilize dynamic programming techniques to solve problems with overlapping subproblems and optimal substructure.	4	-	-	-

National Occupational Standards

Assessment Criteria for Outcomes	Theory Marks	Practical Marks	Project Marks	Viva Marks
<i>Python and Pandas for Data Processing</i>	15	-	-	-
PC18. Explore built-in Python modules and third-party libraries for data processing and analysis, Import and utilize Python modules such as os, sys, and math for system operations, file handling, and mathematical computations.	5	-	-	-
PC19. Install and import external libraries like NumPy and SciPy for scientific computing and advanced mathematical operations, Evaluate the documentation and community support of Python modules to ensure their reliability and compatibility with project requirements.	5	-	-	-
PC20. Understand the Pandas library and its significance in data manipulation and analysis, create Pandas series and Perform data cleaning and tidying operations using Pandas functions like dropna(), fillna(), and replace().	5	-	-	-
<i>Data Analysis and Visualization Mini Projects</i>	12	12	-	-
PC21. Apply tidy data principles to organize and structure data for efficient analysis and visualization, Analyze and visualize time series data using Pandas.	4	4	-	-
PC22. Apply data analysis techniques to real-world datasets related to financial markets, healthcare, or environmental trends, Perform and visualize statistical and data analysis and hypothesis testing to validate research findings and business hypotheses.	4	4	-	-
PC23. Present mini-project findings through written reports and oral presentations to stakeholders and peers.	4	4	-	-
NOS Total	88	12	-	-



National Occupational Standards

National Occupational Standards (NOS) Parameters

NOS Code	ELE/N0801
NOS Name	SIC - Coding and Programming (Essentials of Coding and Programming)
Sector	Electronics
Sub-Sector	
Occupation	Programmer
NSQF Level	4
Credits	3.5
Minimum Educational Qualification & Experience	Pursuing 2nd year of 3-year regular Diploma (after 10th) with NA of experience OR 12th Class (Or Equivalent) with NA of experience OR 10th Class with 3 Years of experience OR Previous relevant Qualification of NSQF Level (Level 3.5) with 3 Years of experience
Version	1.0
Last Reviewed Date	27/08/2024
Next Review Date	27/08/2027
NSQC Clearance Date	27/08/2024
Reference code on NQR	NG-04-EH-02979-2024-V1-ESSC
NQR Version	1.0
CCN Category	1