



Model Curriculum

NOS Name: Essentials of Coding and Programming
OEM Name: SIC – Coding & Programming

NOS Code: ELE/N0801

NOS Version: 1.0

NSQF Level: 4

Model Curriculum Version: 1.0

Electronics Sector Skills Council of India || 155, 2nd Floor ESC House, Okhla Industrial Area – Phase 3,
New Delhi - 110020

Table of Contents

Training Parameters	3
Program Overview.....	4
Training Outcomes.....	4
Compulsory Modules	4
Module Details.....	6
Module 1: Sequential Planning Programming	6
Module 2: List, Tuple. Dictionary and Sequential Data Types	7
Module 3: Algorithm and Data Structures	8
Module 4: Python and Pandas for Data Processing	9
Module 5: Data Visualization Mini Projects.....	10
Annexure	13
Trainer Requirements	13
Assessor Requirements	14
Assessment Strategy	16
References	18
Glossary	18
Acronyms and Abbreviations.....	19

Training Parameters

Sector	Electronics
Sub-Sector	Semiconductor & Components
Occupation	Jr. Programmer (Electronics)
Country	India
NSQF Level	4
Aligned to NCO/ISCO/ISIC Code	NCO-2015/ 2511.0106
Minimum Educational Qualification and Experience	<ol style="list-style-type: none"> 1. 2nd year of 3 year diploma after 10th No Experience required 2. 12th or Equivalent No Experience required 3. 10th Grade pass 3 years relevant experience 4. Previous relevant Qualification of NSQF Level 3.5 3 years relevant experience
Pre-Requisite License or Training	NA
Minimum Job Entry Age	18
Last Reviewed On	27.08.2024
Next Review Date	27.08.2027
NSQC Approval Date	27.08.2024
NOS Version	1.0
Model Curriculum Creation Date	27.08.2024
Model Curriculum Valid Up to Date	27.08.2027
Model Curriculum Version	1.0
Maximum Duration of the Course	104

Program Overview

This section summarizes the end objectives of the program along with its duration.

Training Outcomes:

At the end of the program, the learner should have acquired the listed knowledge and skills:

Compulsory:

- Basic lecture introduces different aspects of Coding and exposure to the current programming world.
- Provides exposure to breaking down complex problems into smaller, manageable parts. Coding and programming not only provide technical skills but also enhance cognitive abilities, foster creativity, and open up numerous career opportunities.
- The lecture series is organized as modules, such as sequential planning, data analysis and visualization, etc

Compulsory Modules:

The table lists the modules and their duration corresponding to the Compulsory NOS.

NOS and Module Details	Theory / Demonstration Duration (In Hours)	Practical/OJT Duration (In Hours)	On-the-Job Training Duration (in hours) (Mandatory)	On-the-Job Training Duration (in hours) (Recommended)	Total Duration (In Hours)
ELE/N0801	80:00	24:00	00:00	00:00	104:00
<i>Module 1: Sequential Planning and Programming</i>	16:00	00:00	00:00	00:00	16:00
<i>Module 2: List, Tuple, Dictionary and Sequential Data Types</i>	16:00	00:00	00:00	00:00	16:00
<i>Module 3: Algorithm and Data Structures</i>	16:00	00:00	00:00	00:00	16:00
<i>Module 4: Python & Pandas for Data Processing</i>	16:00	00:00	00:00	00:00	16:00

<i>Module 5: Data Visualization and project</i>	16:00	24:00	00:00	00:00	40:00
Total Duration	80:00	24:00	00:00	00:00	104:00

Module Details

Module 1: Sequential Planning & Programming

Mapped to ELE/N0801

Terminal Outcomes:

Students will understand the sequential programming & looping techniques to understand the real world.

Duration: 16:00 hrs

Theory - Key Learning Outcomes

- Grasp fundamental programming concepts, including variables, data types, and operators.
- Understand the structure of a simple program, including input, processing, and output.
- Write programs that execute statements sequentially.
- Understand the flow of control in a sequentially executed program.
- Understand and implement various looping constructs such as for, while, and do-while loops.
- Use loops to repeat a block of code multiple times until a condition is met.
- Distinguish between different types of loops and choose the appropriate loop for a given problem.

Practical - Key Learning Outcomes

- Analyze the efficiency of loops and optimize loop performance.
- Understand time complexity basics and how to write efficient iterative code.
- Apply looping techniques in practical scenarios such as data processing, file handling, and user input validation.
- Develop real-world applications that require the use of sequential programming and loops
- Ensure code readability and maintainability by following best practices in coding standards.
- Complete projects and assignments that require the application of sequential programming and looping techniques.
- Demonstrate the ability to integrate loops within larger programs and systems.

Classroom Aids: (If Offline mode)

- Software Installation and Configuration

- Access to Online Resources
- Instructor and TA Support

Tools, Equipment and Other Requirements

- Laptop/Desktop with pre-installed required Software

Module 2: List, Tuple, Dictionary and Sequential Data Types

Mapped to ELE/N0801

Terminal Outcomes:

Students gain a deep understanding and practical proficiency in using these fundamental data structures in programming.

Duration: 16:00 hrs

Theory - Key Learning Outcomes

- Grasp the concept and purpose of sequential data types including lists, tuples, and dictionaries.
- Differentiate between mutable and immutable data structures.
- Understand list operations such as slicing, concatenation, and replication.
- Use built-in list methods like append, remove, pop, sort, and reverse.
- Create and use tuples for storing ordered collections of items.
- Understand the immutability of tuples and the implications for their use.
- Perform operations on tuples including indexing, slicing, and tuple unpacking.
- Understand the concept of key-value pairs and how dictionaries store data.
- Efficiently iterate over lists, tuples, and dictionaries using loops.
- Use list comprehensions and dictionary comprehensions for concise and readable code.
- Analyze the performance implications of different operations on lists, tuples, and dictionaries.
- Understand when to use each data structure based on time and space complexity considerations.
- Utilize Python's built-in functions and modules to work with lists, tuples, and dictionaries effectively.

Practical - Key Learning Outcomes

- Apply the knowledge of lists, tuples, and dictionaries in practical projects and assignments.
- Solve real-world problems using these data structures effectively.
- Debug programs involving lists, tuples, and dictionaries to identify and fix errors.

- Implement error handling to manage exceptions and edge cases when working with these data structures.
- Write clear and well-documented code involving lists, tuples, and dictionaries.

Classroom Aids: (If Offline mode)

- Software Installation and Configuration
- Access to Online Resources
- Instructor and TA Support

Tools, Equipment and Other Requirements

Labs equipped with the following:

- Hardware and software environments well-prepared to support data processing activities. With the right setup, students can effectively learn and apply data manipulation and analysis techniques using sequential programming.

Module 3: Algorithm and Data Structures

Mapped to ELE/N0801

Terminal Outcomes:

It involves equipping students with the knowledge and skills to design, analyze, and implement efficient algorithms and data structures.

Duration: 16:00 hrs

Theory - Key Learning Outcomes

- Understand the importance of algorithm efficiency and performance.
- Learn to design algorithms to solve specific problems.
- Compare and contrast the efficiency of different algorithms for the same problem.
- Understand and implement basic data structures such as arrays, linked lists, stacks, and queues.
- Evaluate the advantages and disadvantages of each data structure.
- Understand the operations and use cases for these advanced data structures.
- Apply these data structures in solving complex problems.
- Implement and analyze various sorting algorithms such as bubble sort, selection sort, insertion sort, merge sort, quicksort, and heapsort.
- Understand the principles and applications of searching algorithms like linear search and binary search.
- Evaluate the performance of different sorting and searching algorithms.
- Understand the concept of recursion and its applications.

- Apply the divide-and-conquer strategy to solve problems and understand its significance in algorithm design.

Practical - Key Learning Outcomes

- Apply the concepts learned to practical projects and real-world scenarios.
- Develop complete programs that effectively use various data structures and algorithms.
- Demonstrate the ability to choose and implement appropriate data structures and algorithms for given problems.

Classroom Aids: (If Offline mode)

- Software Installation and Configuration
- Access to Online Resources
- Instructor and TA Support

Tools, Equipment and Other Requirements

Labs equipped with the following:

- Hardware and software environments well-prepared to support data processing activities. With the right setup, students can effectively learn and apply data manipulation and analysis techniques using algorithm and data processing.

Module 4: Python and Pandas for Data Processing

Mapped to ELE/N0801

Terminal Outcomes: Data processing aims to equip students with the skills needed to manipulate, analyze, and visualize data efficiently.

Duration: 16:00 hrs

Theory - Key Learning Outcomes

- Master the fundamental concepts of Python programming, including variables, data types, control structures, and functions.
- Understand how to work with Python's built-in data structures such as lists, tuples, dictionaries, and sets.
- Learn to import and install Pandas, a powerful data manipulation library in Python.
- Understand and work with Pandas data structures, primarily Series and DataFrame.
- Perform basic operations like indexing, selecting, and filtering data in Pandas.
- Import data from various sources (CSV, Excel, SQL databases, JSON, etc.) into Pandas Data Frames.

Module 5: Data Visualization Mini Projects

Mapped to ELE/N0801

Terminal Outcomes: Aim to ensure that students can apply their knowledge and skills to real-world data problems, resulting in the ability to independently conduct data analysis and create visualizations that communicate insights effectively.

Duration: 16:00 hrs

Theory - Key Learning Outcomes

- Demonstrate the ability to source and collect relevant datasets from various sources such as databases, APIs, web scraping, and publicly available datasets.
- Apply techniques to clean and preprocess raw data, including handling missing values, correcting data types, removing duplicates, and addressing inconsistencies.
- Compute and interpret basic descriptive statistics (mean, median, mode, standard deviation, etc.) to understand the distribution and characteristics of the data.
- Create and interpret basic visualizations (histograms, box plots, scatter plots, etc.) to explore data patterns, trends, and outliers.
- Conduct hypothesis tests to validate assumptions and understand relationships within the data.
- Perform correlation analysis and simple/multiple regression to identify and quantify relationships between variables.
- Use tools such as Matplotlib and Seaborn to create clear, informative static visualizations (bar charts, line graphs, heatmaps, etc.).
- Employ libraries like Plotly or Bokeh to create interactive visualizations that enhance data exploration and presentation.
- Develop interactive dashboards using tools such as Dash or Tableau to present data insights in an accessible and engaging format.
- Develop the ability to craft compelling data-driven narratives that effectively communicate findings to both technical and non-technical audiences.
- Apply best practices in data visualization to ensure clarity, accuracy, and accessibility of visual representations.

Duration: 24:00 hrs

Practical - Key Learning Outcomes

- Apply critical thinking and analytical skills to interpret data results, draw meaningful conclusions, and make data-driven recommendations.
- Demonstrate the ability to approach complex data problems methodically, using appropriate analysis and visualization techniques to find solutions.

- Biometric or manual attendance sheet (stamped by TP) of the trainees during the training period
- Time-stamped & geotagged assessment (Theory + Viva + Practical) photographs & videos

5. Method of verification or validation:

- Surprise visit to the assessment location
- Random audit of the batch
- Random audit of any candidate

6. Method for assessment documentation, archiving, and access

- Hard copies of the documents are stored
- Soft copies of the documents & photographs of the assessment are uploaded / accessed from Cloud Storage
- Soft copies of the documents & photographs of the assessment are stored in the Hard Drives

References

Glossary

Term	Description
Key Learning Outcome	Key learning outcome is the statement of what a learner needs to know, understand and be able to do to achieve the terminal outcomes. A set of key learning outcomes will make up the training outcomes. Training outcome is specified in terms of knowledge, understanding (theory) and skills (practical/OJT application).
Training Outcome	Training outcome is a statement of what a learner will know, understand and be able to do upon the completion of the training
Terminal Outcome	Terminal outcome is a statement of what a learner will know, understand and be able to do upon the completion of a module . A set of terminal outcomes help to achieve the training outcome.
National Occupational Standard	National Occupational Standard specify the standard of performance an individual must achieve when carrying out a function in the workplace
Persons with Disability	Persons with Disability are those who have long-term physical, mental, intellectual, or sensory impairments which in interaction with various barriers may hinder their full and effective participation in society on an equal basis with others

Acronyms and Abbreviations

Term	Description
QF	Qualification File
NSQF	National Skills Qualification Framework
NSQC	National Skills Qualification Committee
NOS	National Occupational Standards
SSC	Skill Sectors Councils
NASSCOM	National Association of Software & Service Companies
NCO	National Classification of Occupations
ISO	International Organization for Standardization
SLA	Service Level Agreement
IT	Information Technology
CRM	Customer Relationship Management
PC	Performance Criteria
PwD	Persons with Disability
SOP	Standard Operating Procedure