School Connect Online

SCO INTERNATIONAL CODING OLYMPIAD

CLASS 10 SYLLABUS

A comprehensive syllabus guide for schools, teachers, parents, and students

Designed from Class 10 coding, AI, and data-processing pathways and aligned with SCO's platform flow for guided preparation, practice, reporting, and future-ready computational thinking.

- exam-ready guidance for Class 10 learners globally
- section-wise pathways across machine learning, NLP, AI bias, Python logic, OpenCV, CNN, and data processing
- preparation roadmap, implementation ideas, and future-benefit framing for coding and AI enrichment

Machine Learning	NLP	AI Bias	Python Logic	OpenCV & CNN
Data Processing	Ethics	Achievers	Coding Skills	AI Projects

School Connect Online

SCO International Coding Olympiad - Class 10 Syllabus Overview

This syllabus supports Class 10 learners in building a structured understanding of coding, artificial intelligence, data processing, responsible AI, and applied computer-vision concepts. The document is intended for students, teachers, schools, and parents who need a quick yet meaningful view of what the Olympiad assesses and how preparation can be guided.

Olympiad progression at this level

Advanced observation, structured reasoning, data interpretation, ethical AI thinking, Python-based problem solving, natural-language processing awareness, and visual-data understanding using OpenCV and CNN foundations.

Exam Information at a Glance

Exam Detail	Class 10 Coding Olympiad Structure
Level	Level 1
Class / Eligibility	Class 10
Duration	60 minutes
Exam Type	Objective-type multiple-choice examination
Number of Questions	50 questions
Content Structure	Five competency blocks including two Achievers sections
Choice of Dates	Cycle 1, Cycle 2, and Cycle 3

Section-Wise Syllabus Blueprint

Section	Learning Block	Core Coverage	Outcome at a Glance
1	Advanced Machine Learning Concepts	Classification, clustering, decision trees, Human Language vs Computer Language	Understands how machine learning systems group, label, and reason from data.
2	Natural Language Processing (NLP)	Basics of language processing in AI; Artificial Intelligence and data processing programs	Recognizes how text is cleaned, tokenized, structured, and converted for AI systems.
3	AI Bias and Fairness	Understanding bias and ethical implications in AI	Explains why fairness, representation, transparency, and responsible deployment matter.

4	Achievers Section 1	Advanced logical puzzles; introduction to Python small NLP projects	Solves code-reading, logic, sequence, and basic NLP-programming problems.
5	Achievers Section 2	AI and data processing programs; OpenCV; CNN in deep learning	Connects images, features, convolution, preprocessing, and AI pipelines.

Recommended question balance

The 50-question structure can be organized as 10 questions per competency block, with progressively higher reasoning demand in both Achievers sections. This keeps the paper balanced across concepts, coding logic, AI ethics, and application-based problem solving.

Chapter Notes and Learning Outcomes

1. Advanced Machine Learning Concepts

This chapter introduces learners to how machines learn from examples and patterns. It focuses on core supervised and unsupervised learning ideas in a way suitable for Class 10, especially classification, clustering, and decision trees.

Learning outcomes:

- Differentiate classification, clustering, and decision-tree approaches.
- Identify labeled and unlabeled data situations.
- Explain model behavior using examples such as spam detection, customer grouping, and rule-based decisions.
- Recognize the risk of overfitting and the need for generalization.

Teaching / practice focus

Use small datasets such as fruits, sports choices, school-library records, or simple customer examples to classify, cluster, and explain patterns.

2. Human Language vs Computer Language

Human language is flexible, expressive, and sometimes ambiguous, while computer language is structured and rule-based. This chapter prepares learners to understand why natural language is challenging for AI systems.

Learning outcomes:

- Compare ambiguity in human sentences with strict programming syntax.
- Explain why context matters in language understanding.
- Identify examples of words with multiple meanings.
- Understand why language data must be processed before AI can use it.

Teaching / practice focus

Ask learners to compare a human sentence, a Python statement, and a chatbot query to see how meaning changes across contexts.

3. Natural Language Processing (NLP)

NLP helps computers process, interpret, and generate human language. The chapter gives learners the foundation needed to understand tokenization, text cleaning, embeddings, sentiment analysis, and simple chatbots.

Learning outcomes:

- Describe tokenization, stop-word handling, stemming, and lemmatization at a beginner level.
- Understand how words can be represented numerically for AI programs.
- Recognize sentiment analysis, named entity recognition, and chatbot use cases.
- Explain the limits of keyword-only language processing.

Teaching / practice focus

Students can analyze simple sentences, split them into words, identify named entities, and compare keyword search with context-aware interpretation.

4. Artificial Intelligence and Data Processing Programs

AI programs depend on reliable data pipelines. This chapter builds the ability to think about data collection, cleaning, transformation, feature extraction, and model-ready input.

Learning outcomes:

- Explain the journey from raw data to usable features.
- Identify missing data, inconsistent labels, noise, and outliers.
- Understand basic tabular and text data transformations.
- Interpret how preprocessing affects model quality.

Teaching / practice focus

Use spreadsheet-like tables, simple text records, and small images to demonstrate cleaning, grouping, counting, and transformation.

5. AI Bias and Fairness

AI systems can produce unfair results when data is incomplete, unbalanced, or shaped by past human decisions. This chapter builds responsible technology awareness for future-ready learners.

Learning outcomes:

- Define AI bias using age-appropriate examples.
- Recognize why diverse and representative data matters.
- Explain fairness concerns in hiring, healthcare, surveillance, education, and finance.
- Discuss privacy, safety, transparency, and human oversight.

Teaching / practice focus

Discuss everyday examples where a rule or system may be unfair because it was designed using incomplete information.

6. Advanced Logical Puzzles

This achievers block develops the logic required for coding and AI. Learners work with patterns, conditions, sequence reasoning, recursion-style thinking, and code-tracing problems.

Learning outcomes:

- Trace a simple algorithm step by step.

- Solve pattern-based and condition-based puzzles.
- Interpret loops, conditions, lists, dictionaries, and functions.
- Use reasoning before selecting a final answer.

Teaching / practice focus

Encourage learners to write intermediate steps, draw flow paths, and verify outputs instead of guessing.

7. Introduction to Python Small NLP Projects

This chapter connects programming with language processing. It introduces learners to practical mini-project thinking using Python structures and beginner NLP tasks.

Learning outcomes:

- Understand Python lists, dictionaries, functions, and basic text operations.
- Read and reason about simple tokenization and frequency-count code.
- Design a small rule-based chatbot or sentiment checker.
- Recognize the limits of simple keyword-based AI programs.

Teaching / practice focus

Mini-projects can include word counters, simple chat-response systems, email-pattern extraction, and positive/negative keyword checkers.

8. Introduction to OpenCV

OpenCV introduces learners to image processing and computer vision. This chapter explains how images are loaded, resized, transformed, filtered, and used in AI pipelines.

Learning outcomes:

- Understand images as pixel data.
- Identify basic operations such as reading, resizing, thresholding, morphology, and video capture.
- Explain why preprocessing helps visual AI systems.
- Connect OpenCV with real-world camera and image-analysis use cases.

Teaching / practice focus

Students can observe how changing brightness, size, edges, or noise affects what a machine might detect.

9. Convolutional Neural Networks (CNN) in Deep Learning

CNNs are important for visual AI. This chapter introduces filters, feature maps, pooling, activation functions, and image-classification pipelines at a conceptual Class 10 level.

Learning outcomes:

- Explain how convolutional filters detect patterns such as edges and shapes.
- Identify roles of convolution, pooling, flattening, dense layers, and softmax output.
- Understand data augmentation and transfer learning at a beginner level.
- Relate CNNs to object detection, medical imaging, traffic-sign recognition, and quality checks.

Teaching / practice focus

Use simple image examples to show how low-level features combine into higher-level recognition.

School Connect Online

Learning Pathway for Students

The Class 10 Coding Olympiad pathway is designed to move learners from concept recognition to problem solving and then to applied reasoning. The goal is not only to remember definitions but to understand how coding, AI, and data systems work together.

Stage 1: Observe and classify

Identify data types, labels, categories, tokens, features, and everyday AI examples.

Stage 2: Process and reason

Apply rules, clean data, trace code, evaluate model outcomes, and compare decisions.

Stage 3: Build and interpret

Read Python-style logic, reason through NLP mini-projects, and interpret OpenCV/CNN pipelines.

Stage 4: Evaluate responsibly

Check fairness, privacy, bias, safety, and the effect of model decisions on people.

Preparation Roadmap

Phase	Focus Area	Suggested Student Practice	Readiness Indicator
Week 1	Machine learning basics	Classification vs clustering, decision-tree examples, labeled vs unlabeled data	Can explain the correct method for a given problem.
Week 2	NLP and data processing	Tokenization, text cleaning, simple frequency counts, sentence interpretation	Can trace basic NLP and Python text-processing logic.
Week 3	Bias, fairness, and ethics	Case discussions on representation, privacy, and responsible AI decisions	Can identify why an AI system may be unfair.
Week 4	Achievers practice	Python puzzles, OpenCV/CNN concepts, mixed application questions	Can solve multi-step reasoning questions with explanation.

Assessment Blueprint

The objective-type paper should test understanding, interpretation, logic, and applied decision-making. Questions may include direct concept checks, code-output reasoning, data-processing scenarios, case studies, and achievers-level application problems.

Competency Block	Suggested Question Share	Assessment Style	Expected Cognitive Demand
Advanced Machine Learning Concepts	Approx. 10 questions	Concept + application	Understand, compare, classify
Natural Language Processing	Approx. 10 questions	Pipeline + code reading	Apply, interpret, reason
AI Bias and Fairness	Approx. 10 questions	Case-based ethics	Evaluate, justify, decide
Achievers Section 1	Approx. 10 questions	Logic + Python NLP	Analyze, trace, solve
Achievers Section 2	Approx. 10 questions	Data processing + OpenCV + CNN	Apply, connect, infer

Question design note

Good Olympiad questions should require more than memory. A strong paper asks learners to compare choices, evaluate a model decision, trace code, and connect a visual/data pipeline to a concept.

Guidance for Students, Teachers, Schools, and Parents

For Students

Focus on understanding the idea behind each term. Practice explaining why an answer is correct and why other options are weaker. For coding questions, trace the code slowly and write intermediate values.

For Teachers

Use small, relatable datasets and simple AI scenarios. Encourage learners to reason about the pipeline: data, preprocessing, model, output, evaluation, and fairness.

For Schools

Use the syllabus as a computational-thinking enrichment pathway. It can support coding clubs, AI awareness sessions, data-literacy activities, and interdisciplinary STEM learning.

For Parents

Support learners by encouraging safe technology use, curiosity, and responsible discussion about AI in daily life. Students benefit from explaining examples in their own words.

Suggested Activities and Mini Projects

- Create a small table of student book preferences and group similar interests using clustering-style reasoning.
- Build a word-frequency table from five sentences and identify repeated words.
- Write a simple rule-based sentiment checker and discuss why it may fail on sarcasm.
- Create a fairness checklist for an AI model used in school admissions, hiring, or healthcare.
- Draw a CNN-style image pipeline: image input, filters, pooling, classifier, output label.
- Compare human language sentences with Python statements and identify why computers need structure.

Quick Glossary

Term	Student-Friendly Meaning
Classification	A machine-learning task that assigns data to predefined labels or categories.
Clustering	A technique that groups similar data points without using predefined labels.
Decision Tree	A model that makes decisions through a tree-like structure of questions and outcomes.
Tokenization	The process of splitting text into smaller units such as words or punctuation marks.
Embedding	A numeric representation of words, images, or data points that helps a model compare meaning or similarity.
Bias	A pattern of unfair or unbalanced output that may occur because of incomplete data or flawed design.
OpenCV	A computer-vision library used for image and video processing.

CNN	A neural-network architecture commonly used for image-recognition tasks.
Data Augmentation	A technique that creates varied training examples, such as rotated or brightened images.
Transfer Learning	A method that uses knowledge from a pre-trained model for a new related task.