

# SCO INTERNATIONAL OLYMPIAD

## CLASS 1 CODING OLYMPIAD SYLLABUS

Solved question paper for schools, teachers, parents, and students

**Designed from Class 1 coding and logic pathways and aligned with SCO's platform flow for guided preparation, practice, reporting, and future-ready digital growth.**

- age-fit coding and computational-thinking practice for Class 3 learners globally
- section-wise pathways across Coding Fundamentals, Logic and Reasoning, Game Creation, Scratch, Algorithms, and Debugging
- answer key, explanations, correction notes, and revision support for confident olympiad preparation

Coding	Logic	Scratch	Games	Robotics Patterns
Algorithms	Debugging	Sprites	Reasoning	Practice

## SCO International Coding Olympiad - Class 1 Syllabus

A clear, school-ready pathway for early-primary coding, AI awareness, patterns, logic, and creative problem solving.

This Class 1 Coding syllabus introduces students to coding as a way of giving clear, ordered instructions. The focus is not on complex programming language syntax; it is on thinking like a young problem solver: observe, order steps, find patterns, make rules, test ideas, and improve them. Students learn through stories, classroom movement, picture-based tasks, block-style commands, simple games, and Olympiad-style reasoning questions.

## Class 1 Coding Learning Pathway



### Purpose of the Class 1 Coding Syllabus

For students: build confidence with simple instructions, patterns, and problem-solving.

For teachers: use short unplugged and screen-based activities that are easy to explain and assess.

For schools: provide a future-ready foundation in computational thinking, safe technology awareness, and logical reasoning.

## Syllabus at a Glance

Five learning chapters designed for progression from basic commands to practical Olympiad preparation.

Chapter	Chapter Name	Core Focus	Student Outcome	Teacher/School Use
1	<b>Basic Concepts of Coding</b>	Commands, sequence, algorithm, input/output, debugging	Follows and arranges simple step-by-step instructions	Introduces coding vocabulary through actions and classroom routines
2	<b>Examples of AI</b>	AI around us, smart devices, voice assistants, recommendations	Recognizes simple AI uses and learns that humans guide technology	Connects coding with real-life smart technology examples
3	<b>Pattern Recognition Games</b>	Shape, colour, number, picture, and movement patterns	Predicts what comes next and explains why	Builds logical reasoning before formal programming
4	<b>Simple Game Creation</b>	Events, movement, rules, win/lose conditions, sprites, debugging	Plans a tiny game/story using ordered blocks or classroom cards	Supports teacher-friendly mini-projects without complex software

<b>5</b>	<b>Olympiad Practice and Latest Scenarios</b>	Question practice, assignments, safe digital behaviour, real-world coding situations	Solves scenario-based questions with confidence and clarity	Prepares schools for structured practice, reporting, and enrichment
----------	---	--	---	---

## Teaching and Learning Approach

Designed for young learners who understand best through movement, pictures, stories, repetition, and short challenges.



<b>Student Experience</b>  Learn by doing: follow arrows, arrange steps, solve pattern puzzles, act as a robot, and improve mistakes.	<b>Teacher Experience</b>  Use short 10-20 minute micro-lessons with board examples, pair work, and quick reflection questions.	<b>School Experience</b>  Build foundational digital literacy with a balanced pathway across logic, creativity, AI awareness, and safe technology use.
---	---	--

# Chapter 1: Basic Concepts of Coding

Students learn that coding means giving clear instructions in the correct order.

## Small Chapter Note

This chapter builds the foundation of computational thinking through daily-life examples: brushing teeth, packing a bag, moving like a robot, and following classroom rules. Students understand that computers and robots do not guess; they follow precise steps.

## Key Vocabulary

- Command - an instruction to do something
- Sequence - steps placed in order
- Algorithm - a planned set of steps
- Input - information given to a computer
- Output - result shown by a computer
- Debugging - finding and fixing mistakes

Learning Outcomes	Classroom Activity Ideas	Practice / Assessment Focus
<ul style="list-style-type: none"> <li>• Identify simple commands such as move, turn, start, stop, and repeat.</li> <li>• Arrange steps in the correct order to complete a task.</li> <li>• Understand algorithm as a step-by-step method to solve a problem.</li> <li>• Recognize simple input and output examples.</li> <li>• Find and fix a mistake in a short instruction sequence.</li> </ul>	<ul style="list-style-type: none"> <li>• Robot Says: one student gives commands while another acts as the robot.</li> <li>• Daily Routine Algorithm: arrange morning routine cards in correct order.</li> <li>• Find the Bug: spot the wrong step in a simple task sequence.</li> <li>• Input and Output: press a key, see a letter appear on the screen.</li> </ul>	<ul style="list-style-type: none"> <li>• Picture-based sequencing questions.</li> <li>• Choose the correct command for a robot movement.</li> <li>• Identify input/output devices from classroom examples.</li> <li>• Explain what is wrong in a short code-like instruction.</li> </ul>

## Latest Scenario / Real-Life Connection

In a smart classroom, a student may press a button to start a display, type a name on a keyboard, or follow step-by-step instructions on a screen. These daily actions help students understand input, output, sequence, and debugging.

## Chapter 2: Examples of AI

Students recognize simple Artificial Intelligence examples in everyday life.



<p><b>Small Chapter Note</b></p> <p>This chapter introduces AI through child-friendly examples such as voice assistants, smart cameras, learning apps, recommendations, maps, and robots. The goal is awareness, not technical machine learning. Students learn that AI uses data and patterns, but humans must use it carefully and responsibly.</p>	<p><b>Key Vocabulary</b></p> <ul style="list-style-type: none"> <li>• AI - technology that can help make decisions or predictions</li> <li>• Smart device - a device that responds or adjusts</li> <li>• Voice assistant - a tool that understands spoken commands</li> <li>• Recommendation - a suggestion based on patterns</li> <li>• Data - information used by a computer</li> <li>• Safety - using technology carefully</li> </ul>
---	--

Learning Outcomes	Classroom Activity Ideas	Practice / Assessment Focus
<ul style="list-style-type: none"> <li>• Recognize simple AI examples such as voice assistants and smart suggestions.</li> <li>• Understand that AI can help people by finding patterns or answering simple questions.</li> <li>• Know that AI is made and guided by people.</li> <li>• Use safe language around AI: ask a trusted adult before sharing personal details.</li> <li>• Compare human thinking and machine help in simple examples.</li> </ul>	<ul style="list-style-type: none"> <li>• AI Around Us: circle AI examples from a picture sheet.</li> <li>• Voice Command Role Play: one student asks, another responds like an assistant.</li> <li>• Good Use or Not: discuss safe and unsafe ways to use smart technology.</li> <li>• Human or Machine: decide which tasks need human care and which can be helped by AI.</li> </ul>	<ul style="list-style-type: none"> <li>• Identify the AI example in a real-life situation.</li> <li>• Choose which smart tool helps in a given scenario.</li> <li>• Answer basic safety questions about asking adults and protecting personal information.</li> <li>• Explain how a recommendation or smart suggestion may be based on repeated choices.</li> </ul>

**Latest Scenario / Real-Life Connection**

Students may see AI when a learning app suggests the next activity, when a phone understands a voice command, or when a map shows a route. The chapter builds early AI literacy with a safe and human-centered mindset.

## Chapter 3: Pattern Recognition Games

Students learn to notice, continue, and explain patterns using pictures, shapes, numbers, and movement.

### Small Chapter Note

Pattern recognition prepares young students for coding because computers often follow repeated rules. Students practice finding what repeats, predicting what comes next, grouping objects, and explaining the rule in simple words.

### Key Vocabulary

- Pattern - something that repeats or follows a rule
- Rule - the reason a pattern works
- Repeat - doing something again
- Sort - put similar things together
- Predict - say what comes next
- Loop - repeated action in coding

Learning Outcomes	Classroom Activity Ideas	Practice / Assessment Focus
<ul style="list-style-type: none"> <li>• Recognize repeating patterns in shapes, colours, numbers, sounds, and actions.</li> <li>• Predict the next item in a simple pattern.</li> <li>• Sort and group objects by a visible rule.</li> <li>• Explain a pattern rule using simple language.</li> <li>• Connect patterns with loops and repeated actions in coding.</li> </ul>	<ul style="list-style-type: none"> <li>• Shape Train: continue a shape pattern on the board.</li> <li>• Sound Pattern: clap-tap-clap-tap and identify the next action.</li> <li>• Colour Sorting: group cards by colour, size, or shape.</li> <li>• Loop Link: show that a repeated pattern can become a repeat command.</li> </ul>	<ul style="list-style-type: none"> <li>• Complete visual and number patterns.</li> <li>• Find the odd item out using a clear rule.</li> <li>• Match a repeated pattern to a loop idea.</li> <li>• Explain the rule behind a classroom pattern.</li> </ul>

### Latest Scenario / Real-Life Connection

Modern learning games often become more useful when students can see patterns. For example, a game may repeat obstacles or levels. Recognizing the pattern helps the student plan the next move logically.

# Chapter 4: Simple Game Creation (Teacher-Friendly Split Chapter)

Students plan and build a very simple game or animated story using basic coding ideas.



<p><b>Small Chapter Note</b></p> <p>This chapter turns concepts into creation. Students do not need advanced coding syntax. They learn how a character can start, move, react to a click, collect an item, earn points, or end the game. Teachers may split this chapter into mini-lessons for easy classroom delivery.</p>	<p><b>Key Vocabulary</b></p> <ul style="list-style-type: none"> <li>• Sprite - a character or object in a game</li> <li>• Event - something that starts an action</li> <li>• Movement - changing position or direction</li> <li>• Score - points earned in a game</li> <li>• Rule - what is allowed or what happens next</li> <li>• Debug - fix the game when it does not work</li> </ul>
---	---

Learning Outcomes	Classroom Activity Ideas	Practice / Assessment Focus
<ul style="list-style-type: none"> <li>• Plan a simple game goal such as collect, avoid, match, or reach.</li> <li>• Use event ideas such as when clicked, when key pressed, or when game starts.</li> <li>• Choose movement commands for a character or object.</li> <li>• Add a simple rule such as if touching, then score or stop.</li> <li>• Test and debug the game idea using teacher guidance.</li> </ul>	<ul style="list-style-type: none"> <li>• Mini Lesson 1: Choose a character and a goal.</li> <li>• Mini Lesson 2: Add start and movement instructions.</li> <li>• Mini Lesson 3: Add a simple rule or score idea.</li> <li>• Mini Lesson 4: Test, fix, and explain the game.</li> </ul>	<ul style="list-style-type: none"> <li>• Identify the event that starts a game action.</li> <li>• Choose the correct movement command for a character.</li> <li>• Match game rules with correct outcomes.</li> <li>• Find why a simple game is not working and suggest a fix.</li> </ul>

<p><b>Latest Scenario / Real-Life Connection</b></p> <p>A child-friendly game may ask a character to collect stars, avoid an obstacle, or reach a finish point. Students learn that every game has instructions, rules, and testing behind it.</p>
--

# Chapter 5: Olympiad Practice, Assignments, and Latest Scenarios in Coding for Kids

Students apply coding concepts to Olympiad-style questions and real-life technology situations.



<p><b>Small Chapter Note</b></p> <p>This chapter prepares students for practice, revision, and confidence-building. It brings together commands, algorithms, loops, conditions, events, AI examples, and pattern puzzles through age-fit assignments and scenario-based questions.</p>	<p><b>Key Vocabulary</b></p> <ul style="list-style-type: none"> <li>• Practice - repeated learning to improve</li> <li>• Scenario - a situation or story-based question</li> <li>• MCQ - multiple-choice question</li> <li>• Revision - review before assessment</li> <li>• Digital safety - careful technology use</li> <li>• Confidence - readiness to try and improve</li> </ul>
--	---

Learning Outcomes	Classroom Activity Ideas	Practice / Assessment Focus
<ul style="list-style-type: none"> <li>• Solve simple MCQ questions based on coding vocabulary and logic.</li> <li>• Use pictures and short stories to identify the correct coding idea.</li> <li>• Practice safe and responsible technology behaviour.</li> <li>• Connect coding with classroom, home, smart-device, and learning-app scenarios.</li> <li>• Build confidence through regular assignments and reflection.</li> </ul>	<ul style="list-style-type: none"> <li>• Weekly Picture Practice: solve five coding picture questions.</li> <li>• One-Minute Explanation: student explains why an answer is correct.</li> <li>• Safe Tech Talk: discuss what information should not be shared online.</li> <li>• Assignment Review: teacher checks patterns, vocabulary, and reasoning.</li> </ul>	<ul style="list-style-type: none"> <li>• Sequencing, pattern, AI example, and game-rule questions.</li> <li>• Short scenario-based reasoning questions.</li> <li>• Simple debugging and odd-one-out tasks.</li> <li>• Vocabulary matching and real-life application questions.</li> </ul>

<p><b>Latest Scenario / Real-Life Connection</b></p> <p>Current coding education for children is moving beyond only screen practice. It now includes logical thinking, unplugged games, safe AI awareness, real-world problem-solving, and teacher-guided reflection.</p>
---

## Visual Learning Supports

Use images, cards, movement, and simple diagrams so every Class 1 learner can participate confidently.

### How young learners meet AI and patterns



#### See a pattern

Shapes, colours, numbers

IF -> THEN

#### Make a choice

If this happens, do that



#### Use AI safely

Ask, check, and learn with a teacher

### Teacher-Friendly Class Flow

#### Unplugged idea

Act it out with movement cards

#### Block logic

Move, turn, repeat, if

#### Mini project

Make a tiny game or story

#### Reflection

Explain the rule and fix bugs

## Suggested Learning Plan and Assignments

Flexible pacing for schools, teachers, and students. Each chapter may be taught through short sessions and practice activities.

Stage	Suggested Duration	Classroom Focus	Assignment Evidence
<b>Start</b>	1-2 weeks	Coding vocabulary, commands, sequence	Oral explanation and picture sequencing worksheet
<b>Build</b>	2-3 weeks	Algorithms, input/output, debugging	Daily routine algorithm and robot instruction card
<b>Explore</b>	2 weeks	AI examples and safe smart technology use	AI-around-us matching sheet
<b>Think</b>	2 weeks	Patterns, loops, sorting, odd-one-out logic	Pattern game worksheet and classroom explanation
<b>Create</b>	2-3 weeks	Simple game design using events and rules	Mini game/story plan with start, movement, score, and fix notes
<b>Prepare</b>	Ongoing	Olympiad-style practice and scenario questions	Practice score review, explanation quality, and correction work

### Assignment Design Note

Assignments should be short, visual, and explanation-friendly. A Class 1 learner should be able to answer by choosing, matching, ordering, drawing a path, or explaining a simple rule in one sentence. Practice should value reasoning, not memorisation.

# Implementation Guidance for Schools and Teachers

A practical classroom-ready checklist for a coding syllabus that is simple, inclusive, and future-ready.



<p><b>For Students</b></p> <p>Observe the problem, arrange steps, try the answer, check the result, and improve the mistake.</p>	<p><b>For Teachers</b></p> <p>Begin with unplugged actions before screen tasks. Use examples from home, school, games, and simple smart devices.</p>
<p><b>For Schools</b></p> <p>Create a consistent practice plan with worksheets, guided activities, safe digital habits, and periodic Olympiad-style assessment.</p>	<p><b>For Parents</b></p> <p>Encourage children to explain daily routines as steps and notice patterns in stories, games, travel, and household tasks.</p>

## Expected Competencies by the End of Class 1

The syllabus is complete when students can use simple coding language, solve logic tasks, and explain their thinking.



<p><b>Student Competency Checklist</b></p> <ul style="list-style-type: none"> <li>• Can follow and create a simple sequence of instructions.</li> <li>• Can identify a pattern and continue it correctly.</li> <li>• Can understand repeat as an early loop idea.</li> <li>• Can recognize an if/then situation in daily life.</li> <li>• Can find a simple mistake in instructions and suggest a fix.</li> <li>• Can name basic AI examples and use smart technology safely with adult guidance.</li> <li>• Can plan a tiny game or story with a character, action, rule, and result.</li> </ul>	<p><b>Olympiad Readiness Indicators</b></p> <ul style="list-style-type: none"> <li>• Answers picture-based coding questions with confidence.</li> <li>• Explains the reason behind a selected answer.</li> <li>• Solves sequence, pattern, odd-one-out, input/output, and debugging questions.</li> <li>• Connects coding terms with classroom and real-life examples.</li> <li>• Completes practice assignments regularly and corrects mistakes after feedback.</li> <li>• Uses age-appropriate digital safety language.</li> </ul>
---	--

## Class 1 Coding Glossary

Short meanings for fast revision and classroom display.

Word	Simple Meaning	Classroom Example
<b>Code</b>	Instructions for a computer or robot	Move forward, turn left, stop
<b>Command</b>	One instruction	Jump once
<b>Sequence</b>	Steps in correct order	Open bag, take book, read page
<b>Algorithm</b>	A full plan of steps	Steps to brush teeth
<b>Loop</b>	Repeat an action	Clap three times
<b>Condition</b>	If something happens, do an action	If it rains, use an umbrella
<b>Event</b>	Something that starts an action	When button is clicked, say hello
<b>Debug</b>	Find and fix a mistake	Robot turns right instead of left
<b>Input</b>	Information given to a computer	Typing a name
<b>Output</b>	Result shown by a computer	Name appears on the screen
<b>AI</b>	Smart technology that can help with decisions or suggestions	Voice assistant or learning app suggestion
<b>Pattern</b>	Repeated or rule-based order	Red-blue-red-blue

## Closing Note for the Learning Community

Coding in Class 1 is a foundation for logic, creativity, safe technology awareness, and future-ready confidence.

### SCO International Coding Olympiad - Class 1

This syllabus helps young learners begin their coding journey through clear instructions, simple logic, patterns, games, and safe AI awareness. Teachers and schools can use it as a structured pathway for classroom teaching, Olympiad preparation, practice assignments, and future digital confidence.