School Connect Online

SCO INTERNATIONAL CODING OLYMPIAD

CLASS 8 SAMPLE PAPER

Sample Practice Paper Set B | With Answer Key and Explanations

Official SCO cover format with academic, ready question layout.

- Designed for Class 8 learners developing programming, debugging, web, data and game logic skills.
- Compact question numbering, clean question blocks, answer key and explanations for website download.
- Aligned to middle-school computational thinking, C++/Kotlin/Python/XML, web basics, game logic and responsible coding practice.

Programming	C++	Kotlin	Python	Turtle	
Scratch	JavaScript	SQL	Game Logic	SCO	

School Connect Online

Section 1 - Programming Basics

Q1. What will be the output of this C program?

```
#include <stdio.h>
int main() {
int a = 5;
printf("%d %d", a++, ++a);
return 0;
}
```

A. 5 6
B. 6 6
C. 5 7
D. Undefined behavior

Q2. What is the correct syntax to declare a pointer in C?

A. `int ptr;`
B. `int *ptr;`
C. `ptr *int;`
D. `pointer<int> ptr;`

Q3. What will the following program print?

```
#include <stdio.h>
int main() {
int arr[] = {10, 20, 30, 40};
printf("%d", *(arr + 2));
return 0;
}
```

A. 10
B. 20
C. 30
D. 40

Q4. What does sizeof() return when applied to an integer variable?

A. Size of the integer in bits
B. Size of the integer in bytes
C. Always 4
D. Always 8

Q5. What is the purpose of fopen() in C?

A. `Opens a file for reading and writing`
B. Creates an array
C. Declares a function
D. Runs an infinite loop

Q6. What does XML stand for?

A. Extra Markup Language
B. eXtensible Markup Language
C. Externally Managed Language
D. Extensible Managing Layout

Q7. Which of the following is NOT a feature of XML?

A. Human-readable
B. Self-descriptive
C. Fixed set of tags
D. Used to structure data

Q8. What is the root element in this XML document?

```
<game>
<player>John</player>
<score>2500</score>
</game>
```

A. game
B. player
C. score
D. None of the above

Q9. How do you write an XML comment?

A. // This is a comment
B. `<!-- This is a comment -->`
C. /* This is a comment */
D. `{ This is a comment }`

Q10. What is the purpose of an XML Schema (XSD)?

A. To define the structure and data types of XML documents
B. To store images
C. To write Python code
D. To add styles like CSS

Q11. What will the following Python Turtle code do?

```
import turtle
t = turtle.Turtle()
```

t.circle(50)

A. Draw a circle with radius 50
B. Move forward by 50 units
C. Draw a square
D. Show an error

Q12. What will happen if you run the following Turtle program?

```
import turtle
t = turtle.Turtle()
```

t.forward(100)
t.right(90)
t.forward(100)

A. The turtle moves 100 steps forward and turns right
B. The turtle moves in a circle
C. The turtle moves randomly
D. The program crashes

Q13. What does this Turtle code do?

```
import turtle
t = turtle.Turtle()
for i in range(4):
```

t.forward(100)
t.right(90)

- **A.** Draws a circle
- **B.** Draws a square
- **C.** Moves in a zigzag pattern
- **D.** Moves forward without turning

Q14. What is the output of this Turtle code?

```
import turtle
t = turtle.Turtle()
```

t.color("blue")
t.forward(50)

- **A.** Turtle moves forward and changes to blue
- **B.** Turtle moves forward and changes to red
- **C.** Turtle moves backward
- **D.** Turtle moves forward but remains black

Q15. What does t.speed(0) do in Turtle programming?

- **A.** Stops the turtle
- **B.** Makes the turtle move fastest
- **C.** Slows down the turtle
- **D.** Moves the turtle in random directions

Section 2 - Advanced Coding Concepts

Q16. Which of the following is NOT a valid data type in C?

- **A.** int
- **B.** float
- **C.** string
- **D.** double

Q17. What will be the output of the following C code?

```
#include <stdio.h>
int main() {
int a = 5;
printf("%d", a++);
return 0;
}
```

- **A.** 5
- **B.** 6
- **C.** Error
- **D.** Undefined

Q18. Which loop in C executes at least once, even if the condition is false?

A. `for loop`
B. `while loop`
C. `do-while loop`
D. `if statement`

Q19. What is the correct syntax to declare a function in C?

A. `function sum(int a, int b) { return a + b; }`
B. `void sum(int a, int b) { return a + b; }`
C. `int sum(int a, int b);`
D. `declare sum(int a, int b);`

Q20. What will be the output of the following C program?

```
#include <stdio.h>
int main() {
int arr[] = {1, 2, 3, 4, 5};
printf("%d", arr[3]);
return 0;
}
```

A. 1
B. 2
C. 3
D. 4

Q21. Which of the following is true about XML?

A. XML tags are predefined
B. `XML is used for data storage and transfer`
C. XML must be compiled before running
D. XML is a programming language

Q22. Which of the following is a valid XML element?

A. `<name>John Doe</name>`
B. `<1name>John Doe</1name>`
C. `<name>John Doe`
D. `<name John Doe />`

Q23. What is an XML attribute used for?

A. Storing extra information about an element
B. Replacing the element name
C. Writing JavaScript code
D. Defining a database

Q24. Which of the following is a difference between XML and HTML?

A. `XML is used for web page design, while HTML is for data storage`
B. `XML allows custom tags, while HTML has predefined tags`
C. `XML ignores errors, while HTML enforces strict rules`
D. XML and HTML are identical

Q25. Which language is commonly used to parse XML data?

A. C++
B. Python
C. SQL
D. HTML

Q26. What will be the output of the following C program?

```
#include <stdio.h>
void printPattern(int n) {
for (int i = 1; i <= n; i++) {
for (int j = 1; j <= i; j++) {
printf("%d", j);
}
for (int k = i - 1; k >= 1; k--) {
printf("%d", k);
}
printf("\n");
}
}
int main() {
printPattern(4);
return 0;
}
```

A. 1
121
12321
1234321

B. 1
22
333
4444

C. 1
12
123
1234

D. 1
12
1234
123456

Q27. Given the following XML document, what will be the output of the Python XML parser?

```
<students>
<student id="1">
<name>Alice</name>
<marks>85</marks>
</student>
<student id="2">
<name>Bob</name>
<marks>90</marks>
</student>
</students>
import xml.etree.ElementTree as ET
xml_data = '''<students>...</students>'''  # XML data here
root = ET.fromstring(xml_data)
for student in root.findall('student'):
name = student.find('name').text
marks = int(student.find('marks').text)
if marks > 87:
print(name)
```

A. Alice

B. Bob

C. Alice
Bob

D. 85
90

Q28. What will be the output of the following C program considering memory allocation?

```
#include <stdio.h>
#include <stdlib.h>
void allocateMemory(int **ptr) {
*ptr = (int *)malloc(3 * sizeof(int));
(*ptr)[0] = 10;
(*ptr)[1] = 20;
(*ptr)[2] = 30;
}
int main() {
int *arr;
allocateMemory(&arr);
arr[1] = 50;
printf("%d %d %d", arr[0], arr[1], arr[2]);
free(arr);
return 0;
}
```

A. 10 50 30

B. Garbage Garbage Garbage

C. Segmentation Fault

D. 10 20 30

Q29. What shape is drawn by the following Python Turtle code?

```
import turtle
t = turtle.Turtle()
t.speed(0)
for i in range(36):
for j in range(4):
t.forward(100)
t.right(90)
t.right(10)
turtle.done()
```

A. A single square
B. A spiral of squares
C. A circular pattern of squares
D. A triangle pattern

Q30. What will be the output of the following C program using #define preprocessor directives?

```
#include <stdio.h>
#define X 10
#define Y X + 5
int main() {
int result = Y * 2;
printf("%d", result);
return 0;
}
```

A. 30
B. 20
C. 25
D. Error: Undefined macro

Q31. What is the best way to detect collision between two rectangular objects in a 2D game written in C?

A. `Checking if their x-coordinates are equal.`
B. `Checking if their widths are equal.`
C. Using the Axis-Aligned Bounding Box (AABB) method.
D. Using only the player's position.

Q32. How can XML be used in game development?

A. To store and retrieve game configuration settings.
B. To execute in-game physics calculations.
C. To write the main game loop.
D. To handle player movement in real-time.

Q33. In Python's Turtle module, what happens when the following code is executed?

```
import turtle
t = turtle.Turtle()
t.speed(0)
for i in range(50):
t.forward(i * 5)
t.right(144)
```

A. A spiral pattern appears.
B. A star-like shape is drawn.
C. A random path is created.
D. The turtle moves in a straight line.

Q34. Which function is commonly used in C programming to generate a random game event, such as a random enemy spawn location?

A. `time()`
B. `rand()`
C. srand(time(NULL))
D. `random.seed()`

Q35. What is the correct way to store a player's high score in an XML file?

A.
```
<game>
<player name="Alice" score="1200"/>
</game>
```
B.
```
<game>
<player>
<name>Alice</name>
<score>1200</score>
</player>
</game>
```
C.
```
<game>
<player name: "Alice" score: "1200"/>
</game>
```
D.
```
<game>
<player>
<score name="Alice">1200</score>
</player>
</game>
```

Q36. What is event-driven programming in game development?

A. A programming style where the game runs only when the player clicks buttons.
B. A programming approach where the game continuously updates itself.
C. A programming paradigm where user actions (events) trigger responses.
D. A method where all functions run in a fixed order.

Q37. Which of the following is the best way to move a Turtle left when the left arrow key is pressed?

A. t.left(90) only, with no event binding
B. `Use a key event such as screen.onkeypress(move_left, "Left") after defining move_left and calling screen.listen().`
C. t.backward(90)
D. t.speed(-90)

Q38. Which AI algorithm is best for finding the shortest path efficiently in a 2D game map with obstacles?

A. Bubble Sort
B. Depth-First Search
C. Breadth-First Search only
D. A* Algorithm

Q39. Smart AI Enemy Behavior in a Game

```
A game developer is implementing an AI enemy that chases the player efficiently in a 2D maze while avoiding obstacles. Which algorithm is the best choice?
```

A. Dijkstra's Algorithm - reliable but explores broadly without a goal heuristic.
B. A* Algorithm - uses a heuristic to find a fast route efficiently.
C. Depth-First Search - may explore long wrong paths first.
D. `Breadth-First Search - good for unweighted grids but less focused than A*.`

Q40. Precise Hit Detection in a Fast-Paced Game

A game developer is creating a fast-paced shooter game where bullets move at high speeds across the screen. The bullets should hit the target accurately, but sometimes they seem to "pass through" thin objects without registering a hit.
Which of the following techniques is the best solution to accurately detect collisions at high speed?

A. Axis-Aligned Bounding Box (AABB) Collision – Uses rectangular bounds to check overlap.
B. Pixel-Perfect Collision – Checks the exact overlapping pixels of two objects.
C. Raycasting – Simulates a line from the bullet's position to predict collisions before they happen.
D. Frame-Based Collision Detection – Only checks collisions at each frame update.

Q41. In a C-based game, the main game loop is structured as follows:

```
while (running) {
processInput();
updateGame();
renderGraphics();
}
What will happen if the updateGame() function has an infinite loop?
```

A. The game will run normally but slow down.
B. The game will freeze and stop responding.
C. The graphics will continue updating, but input will be delayed.
D. The game will crash immediately without any error message.

Q42. Why is XML commonly used in game development for storing level designs?

A. XML allows real-time execution of game logic.
B. XML can be used to store structured data like game levels in a human-readable format.
C. XML is faster than binary data storage.
D. `XML is required for all modern game engines.`

Q43. Which of the following best describes Axis-Aligned Bounding Box (AABB) collision detection in games?

A. `Checks if the bounding boxes of two objects overlap.`
B. Uses raycasting to detect precise object collisions.
C. `Relies on exact pixel-matching for collision accuracy.`
D. Requires machine learning models to determine collision points.

Q44. In Turtle Programming (Python), which command is used to make the turtle move without drawing a line?

A. `turtle.forward(50)`
B. `turtle.penup()`
C. `turtle.pendown()`
D. `turtle.move(50)`

Q45. A game developer wants to simulate gravity in a 2D platformer using C. Which of the following is the best way to update an object's vertical position each frame?

A. `positionY += speed;`
B. `positionY += gravity * time;`
C. `positionY = speed / gravity;`
D. `positionY = positionX * positionY;`

Q46. A developer is deciding between XML and JSON for saving game data. Why might JSON be better?

A. JSON is always smaller in file size.
B. JSON supports storing images directly in text.
C. JSON has a simpler, lightweight structure compared to XML.
D. `JSON is required for saving game data.`

Q47. A developer wants to create a random-moving turtle in Python. Which function should be used?

A. `turtle.random()`
B. `random.choice()`
C. `turtle.random_move()`
D. `turtle.randint()`

Q48. Why would a game developer use multi-threading in C for a game?

A. To allow multiple tasks/threads such as physics, AI or loading to run concurrently where appropriate.
B. To make the game easier to debug automatically.
C. To ensure the game never crashes.
D. To increase graphics quality without code changes.

Q49. What will the following Python Turtle code output?

```
import turtle
t = turtle.Turtle()
for i in range(3):
```
t.forward(100)
t.left(120)

A. A triangle
B. A square
C. A circle
D. A hexagon

Q50. A game developer is using an XML file to define the map layout of a game. What file format is commonly used for storing tile-based maps in game development?

A. .exe
B. .tiled
C. .tmx
D. .png

School Connect Online

Answer Key

Q	Ans	Correct Option
1	D	Undefined behavior
2	B	int *ptr;
3	C	30
4	B	Size of the integer in bytes
5	A	Opens a file for reading and writing
6	B	eXtensible Markup Language
7	C	Fixed set of tags
8	A	game
9	B	<!-- This is a comment -->
10	A	To define the structure and data types of XML documents
11	A	Draw a circle with radius 50
12	A	The turtle moves 100 steps forward and turns right
13	B	Draws a square
14	A	Turtle moves forward and changes to blue
15	B	Makes the turtle move fastest
16	C	string
17	A	5
18	C	do-while loop
19	C	int sum(int a, int b);
20	D	4
21	B	XML is used for data storage and transfer
22	A	<name>John Doe</name>
23	A	Storing extra information about an element
24	B	XML allows custom tags, while HTML has predefined tags
25	B	Python
26	A	1 121 12321 1234321
27	B	Bob
28	A	10 50 30
29	C	A circular pattern of squares
30	B	20
31	C	Using the Axis-Aligned Bounding Box (AABB) method.
32	A	To store and retrieve game configuration settings.
33	B	A star-like shape is drawn.
34	B	rand()
35	B	<game> <player> <name>Alice</name> <score>1200</score> </player> </game>
36	C	A programming paradigm where user actions (events) trigger responses.
37	B	Use a key event such as screen.onkeypress(move_left, "Left") after defining move_left and calling screen.listen().
38	D	A* Algorithm
39	B	A* Algorithm - uses a heuristic to find a fast route efficiently.
40	C	Raycasting – Simulates a line from the bullet's position to predict collisions before they happen.
41	B	The game will freeze and stop responding.
42	B	XML can be used to store structured data like game levels in a human-readable format.
43	A	Checks if the bounding boxes of two objects overlap.
44	B	turtle.penup()
45	B	positionY += gravity * time;
46	C	JSON has a simpler, lightweight structure compared to XML.
47	B	random.choice()
48	A	To allow multiple tasks/threads such as physics, AI or loading to run concurrently where appropriate.

Q	Ans	Correct Option
49	A	A triangle
50	C	.tmx

Detailed Explanations

Q1. Answer D: Undefined behavior

The expression printf("%d %d", a++, ++a); modifies a twice without a sequence point, leading to undefined behavior.

Q2. Answer B: int *ptr;

The * indicates that ptr is a pointer to an integer in C.

Q3. Answer C: 30

arr + 2 moves two positions ahead, so *(arr + 2) accesses the third element (30).

Q4. Answer B: Size of the integer in bytes

sizeof(int) typically returns 4 bytes on most systems, but it depends on the architecture.

Q5. Answer A: Opens a file for reading and writing

fopen(filename, mode) opens a file in the specified mode ("r", "w", "a", etc.).

Q6. Answer B: eXtensible Markup Language

XML is used to store and transport structured data.

Q7. Answer C: Fixed set of tags

Unlike HTML, XML allows users to define their own tags.

Q8. Answer A: game

The root element is the outermost tag enclosing all elements.

Q9. Answer B: <!-- This is a comment -->

XML uses <!-- --> for comments.

Q10. Answer A: To define the structure and data types of XML documents

XML Schema (XSD) ensures XML data follows specific rules and formats.

Q11. Answer A: Draw a circle with radius 50

circle(50) draws a circle of radius 50 pixels.

Q12. Answer A: The turtle moves 100 steps forward and turns right

forward(100) moves the turtle 100 pixels, and right(90) turns it right by 90 degrees.

Q13. Answer B: Draws a square

The loop moves 100 steps and turns 90° four times, forming a square.

Q14. Answer A: Turtle moves forward and changes to blue

t.color("blue") changes the pen color to blue.

Q15. Answer B: Makes the turtle move fastest

t.speed(0) sets the fastest drawing speed in Turtle.

Q16. Answer C: string

C does not have a built-in string data type. Instead, strings are handled as arrays of characters (char arrays).

Q17. Answer A: 5

The a++ prints the value first (5) and then increments a after execution.

Q18. Answer C: do-while loop

The do-while loop executes at least once because the condition is checked after execution.

Q19. Answer C: int sum(int a, int b);

The correct function declaration in C must include return type, function name, and parameter list.

Q20. Answer D: 4

Arrays in C are zero-indexed, so arr[3] refers to the fourth element, which is 4.

Q21. Answer B: XML is used for data storage and transfer

XML is used to store and transfer structured data, and its tags are user-defined.

Q22. Answer A: <name>John Doe</name>

XML elements must have matching opening and closing tags, and tag names cannot start with numbers.

Q23. Answer A: Storing extra information about an element

Attributes store additional information about an XML element inside its opening tag.

Q24. Answer B: XML allows custom tags, while HTML has predefined tags

XML allows users to define their own tags, while HTML has predefined tags.

Q25. Answer B: Python

Python has libraries like xml.etree.ElementTree and BeautifulSoup for parsing XML.

Q26. Answer A: 1 121 12321 1234321

The outer loop runs from 1 to n=4.
The first inner loop prints numbers from 1 to i.
The second inner loop prints numbers from i-1 down to 1.
This creates a symmetrical pattern with numbers.

Q27. Answer B: Bob

The XML file is parsed, and findall('student') gets all <student> nodes.
The if marks > 87: condition filters students who scored more than 87.
Only Bob has 90, so Bob is printed.

Q28. Answer A: 10 50 30

allocateMemory(&arr); passes the address of arr, so arr points to dynamically allocated memory.
arr[1] = 50; modifies the second element.
printf("%d %d %d", arr[0], arr[1], arr[2]); prints 10 50 30.

Q29. Answer C: A circular pattern of squares

The outer loop (range 36) makes 36 iterations.
The inner loop (range 4) creates a square.
The t.right(10) rotates the turtle slightly after each square, forming a circular pattern of squares.

Q30. Answer B: 20

The macro expands result = Y * 2 into result = 10 + 5 * 2. Multiplication happens before addition, so the result is 20.
Using #define Y (X + 5) would produce 30.

Q31. Answer C: Using the Axis-Aligned Bounding Box (AABB) method.

The AABB method checks if two rectangles overlap by comparing their x, y coordinates and their width/height values.
This is one of the most efficient ways to detect collisions in 2D games.

Q32. Answer A: To store and retrieve game configuration settings.

XML is commonly used to store game data such as level settings, character attributes, and player progress in a structured way.

Q33. Answer B: A star-like shape is drawn.

The turtle moves forward increasingly and turns by 144 degrees, forming a repeating star pattern.

Q34. Answer B: rand()

rand() generates pseudo-random values in C. srand(time(NULL)) is commonly used to seed the generator so the sequence changes across runs.

Q35. Answer B: <game> <player> <name>Alice</name> <score>1200</score> </player> </game>

Option B correctly nests <name> and <score> inside <player>, making it properly structured XML.

Q36. Answer C: A programming paradigm where user actions (events) trigger responses.

In event-driven programming, keyboard presses, mouse clicks, and game actions trigger specific functions.

Q37. Answer B: Use a key event such as screen.onkeypress(move_left, "Left") after defining move_left and calling screen.listen().

Keyboard controls are event-driven. In Turtle, a screen-level key binding and listen() call are used so the function runs when the key is pressed.

Q38. Answer D: A* Algorithm

A* uses path cost plus a heuristic estimate to find shortest paths efficiently, which makes it popular for game pathfinding.

Q39. Answer B: A* Algorithm - uses a heuristic to find a fast route efficiently.

A* combines the path cost used by Dijkstra-style search with a heuristic such as Manhattan distance, allowing the enemy to find efficient routes through a maze.

Q40. Answer C: Raycasting – Simulates a line from the bullet's position to predict collisions before they happen.

Raycasting is the best solution because it traces the bullet's path between frames and detects collisions even if the object moves too fast to be detected by regular frame-based methods.
AABB Collision Detection works well for slow-moving objects but fails for high-speed bullets that may "skip" past objects between frames.
Pixel-Perfect Collision is accurate but too slow for real-time gameplay and doesn't solve the issue of fast movement.
Frame-Based Collision Detection only checks collisions once per frame, which causes fast-moving objects to sometimes miss collisions entirely (this is called the tunneling problem).
Real-World Use:
Raycasting is widely used in FPS games like Call of Duty and CS:GO to calculate bullet trajectory and detect hits accurately.
It's also used in racing games for detecting track boundaries and in 3D games for AI vision mechanics!

Q41. Answer B: The game will freeze and stop responding.

The main loop cycles through input processing, game updates, and rendering. If updateGame() has an infinite loop, the program never returns to process input or render frames, causing the game to freeze indefinitely.

Q42. Answer B: XML can be used to store structured data like game levels in a human-readable format.

XML is widely used for storing game settings, level designs, and character data because it is structured, flexible, and easy to read. However, it is not required for all engines, and binary formats are often faster.

Q43. Answer A: Checks if the bounding boxes of two objects overlap.

AABB collision detection is a fast and efficient way to check if two objects collide by comparing their bounding rectangles. It is commonly used in 2D platformer games.

Q44. Answer B: turtle.penup()

The penup() function lifts the pen so the turtle moves without drawing a line, whereas pendown() resumes drawing.

Q45. Answer B: positionY += gravity * time;

Gravity is an acceleration, so the correct formula updates the Y-position based on gravity over time.

Q46. Answer C: JSON has a simpler, lightweight structure compared to XML.

JSON is more compact and faster to parse than XML, making it a popular choice for saving player progress.

Q47. Answer B: random.choice()

random.choice() is used to randomly select a direction or step for turtle movement.

Q48. Answer A: To allow multiple tasks/threads such as physics, AI or loading to run concurrently where appropriate.

Multithreading can help separate work such as background loading, AI or physics, but it must be designed carefully to avoid race conditions.

Q49. Answer A: A triangle

The loop repeats 3 times, moving 100 units forward and turning 120 degrees, forming a triangle.

Q50. Answer C: .tmx

.tmx is the Tiled Map XML format, widely used for storing 2D game maps.