The Lab

Junior Program

THE LAB

junior

Foundation Term 2
Home Workbook

Preface

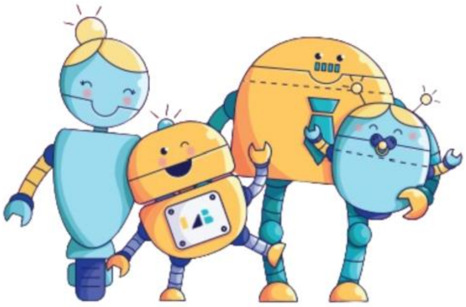
Version Oct 2025

Welcome to The Lab Singapore's workbook—a guide designed to spark creativity, nurture logical thinking, and develop essential coding skills.

At The Lab, we believe that every child can achieve remarkable things when given the right tools and guidance. Our passionate educators, driven by a love for teaching and innovation, are dedicated to empowering students to solve real-world problems through coding and computational thinking.

This workbook reflects our commitment to making learning engaging, meaningful, and fun. It is more than a collection of exercises—it's a step towards building confidence, resilience, and a lifelong love for learning.

Let's embark on this exciting journey together!



A Friendly Note to Parents & Students

This workbook is designed to help our students revise and practise the concepts taught in class. Many of these ideas are new, so it's completely normal for students to need repetition and extra practice before they fully understand them. Not getting it the first time is part of the learning journey — and absolutely okay!

Please remember that this workbook is not a test and it does not reflect how well your child is performing in class or how well we teach. It simply gives students a chance to try things on their own at home, at their own pace.

To keep things encouraging, here's how we look at the scores:

✓ Scoring (Just for Practice!)

50% and above – Great job!

Your child has understood the main ideas.

Below 50% – Keep practising!

This is not a failure. It just shows which areas are still new or need a bit more time — and we'll continue to guide them in class.

At The Lab, we believe learning should feel positive, safe, and joyful. We hope this workbook supports your child as they grow, explore, and discover new concepts!

Answer Key

Conditionals with Touch Sensor

Coding Concepts

1. Answer: B. The Forever block keeps repeating the code inside it, the robot will keep checking and running the sensor action nonstop as long as the program is running.

2. Answer: C. The Forever block makes the robot keep repeating the sensor check nonstop, so it can detect any changes continuously.

3. Answer: C. The robot needs to keep checking the color sensor and adjusting its movement nonstop, and the Forever block makes it follow the line continuously as long as the program is running.

4. Answer: C. Without a Forever block, the sensor code runs only once and then stops, so the robot will not keep sensing continuously.

5. Answer: B. A touch sensor's job is to detect when it is pressed or released, so it can tell the robot that something has touched it.

6. Answer: B. When the touch sensor is pressed, it sends a signal to the robot to let it know that it has been touched.

7. Answer: D. The robot can be programmed to respond with specific actions. For example: if the touch sensor is press, then the printer will move.

8. Answer: B. The robot will move for 1 rotation once because the motor block is outside the forever block and the forever block is empty and does nothing.

9. Answer: A. The forever block is not doing anything because there's no code inside the forever block.

10. Answer: A. Both the motor and the touch sensor are set to use port A, and one port cannot control two devices at the same time, so the program cannot run.,

11. Answer: D. It may or may not run because the code is missing the forever block. Without the forever block, the sensor cannot detects continuously.

12. Answer: D. It may or may not run because the forever block does not include touch sensor block. All the code must be inside forever.

13. Answer: C. Placing the forever block outside the if block makes the program keep checking the sensor continuosly so it can detect the press anytime.

14. Answer: B. The "pressed" means the touch sensor has been pushed down, so the robot detects that a press has happened..

15. Answer: B. The "hard-pressed" means the sensor is pushed down firmly, so it has detected a strong press

More Conditionals with Touch Sensor

Coding Concepts

1. Answer: B. An “if” statement checks a condition and lets the program choose what to do next, helping it make decisions..

2. Answer: B. An “If” statement checks a condition. In this question, this “if” statement checks the weather condition (sunny or not) and helps you decide whether you need a raincoat.

3. Answer: B. The “forever” block is important when using sensor in Lego spike is because it keeps the sensor checking continuously, so the robot never stops sensing.

4. Answer: C. Touch sensor is important because it will respond to physical interactions. A touch sensor lets the robot detect when it is pressed or bumped.

5. Answer: B. This statement directly checks the player’s score, which is the condition you need to decide what happens in the game.

6. Answer: C. This statement directly checks whether the number is even or odd.

7. Answer: A. The robot only needs to be told once which motors it should use, so the set movement block should stay outside the forever block.

8. Answer: C. Port A is already used by a motor (A+B movement), so the touch sensor cannot also be placed in port A at the same time.

9. Answer: B. Set movement block should come first because we want the robot to know which ports the motors are connected to. The movement block should be placed inside the “if” block if we want the robot to move only when the button is pressed.

10. Answer: A. Everything in the code is set up correctly.

11. Answer: B. The movement block is inside the “if” statement, so the motors will only run when the touch sensor is pressed.

12. Answer: D. The touch sensor is plugged into port A, and by default the movement block also tries to use motor A. The robot cannot run both on the same port, so nothing will happen when you press the sensor

13. Answer: C. The condition inside the “if” block specifically checks whether the left button is pressed, so the motors will move only when the left button is pressed.

14. Answer: D. The condition inside the “if” block specifically checks whether the right button is pressed, so the motors will move only when the right button is pressed.

15. Answer: A. The motors are supposed to move when the right button is pressed, so saying they will not move when the right button is pressed is incorrect.

Conditionals with Brick Button

Coding Concepts

1. Answer: B. When using "if-else" statement and "if" part is true, the code in "if" part will run.

2. Answer: B. The else part in "if-else" statement provides an alternative action if the "if" condition is false.

3. Answer: C. If "if" statement is false and there's no "else" part, the robot will do nothing and go to next code if any.

4. Answer: C. The "if-else" statement in programming to give different instruction or alternative action to a robot based on a condition.

5. Answer: B. Using "if-else" statement and "if" part is false, it will run the "else" part.

6. Answer: C. This statement directly checks whether it is morning, which is the condition needed for the program to decide between saying "Good morning" or "Good night.".

7. Answer: C. The "else" part in "if-else" statement is to provide alternative action if the "if" condition is false.

8. Answer: C. This statement directly checks whether the temperature is above 25 degrees Celsius, which is the condition needed for the program to decide when to play music.

9. Answer: C. If the "if" condition is true, we will execute the code inside the "if" block.

10. Answer: A. The code plays the crazy laugh in the "else" part, which happens when the touch sensor is not released, meaning it is pressed.

11. Answer: B. Crazy laugh sound is placed in the "else" section, which runs only when the touch sensor is not pressed (released).

12. Answer: A. The "move" block is inside the "if" condition, so when the touch sensor is not pressed (released), the robot will move.

13. Answer: B. The "move" block only runs when the sensor is pressed, so when the sensor is not pressed, the program goes to the "else" part and plays the crazy laugh sound.

14. Answer: C. The "else" block runs whenever the touch sensor is *not pressed*, and that includes both *released* and *never touched at all*, so you don't need to do anything.

15. Answer: A. The crazy laugh sound is in the "else" part, which runs when the sensor is not released, meaning the sensor must be pressed for the laugh to play.

Conditionals with Ultrasonic Sensor

Coding Concepts

1. Answer: D. “if–else” lets the program choose between two actions depending on whether a condition is true or false, allowing it to make different decisions.

2. Answer: C. This statement directly checks whether it is daytime, which is the condition needed for the robot to decide when to move forward or stop.

3. Answer: C. This statement directly checks whether the spacebar is pressed, which is the condition needed for the character to jump in the game.

4. Answer: A. The condition checks whether the distance is less than 100 cm, so the robot will move only when the ultrasonic sensor detects an obstacle less than 100 cm.

5. Answer: B. Because 90 cm is less than 100 cm, the condition becomes true, so the robot will move.

6. Answer: B. Because 110 cm is not less than 100 cm, the condition is false, so the robot will not move.

7. Answer: B. The condition checks whether the distance is more than 100 cm, so the robot will move only when the ultrasonic sensor detects an obstacle more than 100 cm.

8. Answer: B. Because 90 cm is not more than 100 cm, the condition is false, so the robot will not move.

9. Answer: D. Because 110 cm is more than 100 cm, the condition becomes true, so the robot will move.

10. Answer: C. Because 150 cm is more than 100 cm, the condition becomes true, so the robot will move.

11. Answer: B. Because 90 cm is not equal to 100 cm, the condition is false, so the robot will not move.

12. Answer: B. Because 110 cm is not equal to 100 cm, the condition is false, so the robot will not move.

13. Answer: D. Because 110 cm is equal to 100 cm, the condition becomes true, so the robot will move.

14. Answer: C. The condition checks whether the distance is equal to 100 cm, so the robot will move only when the ultrasonic sensor detects an obstacle at 100 cm.

15. Answer: C. The condition checks whether the distance is equal to 100 cm, so it will do nothing when the distance is less than 100 cm (I) or more than 100 cm (II).

More Conditionals with Ultrasonic Sensor

Coding Concepts

1. Answer: B. Because the ultrasonic sensor can detect how far away obstacles are, helping the robot avoid bumping into walls while moving along a path.

2. Answer: A. The arrow on the motor block is pointing to the right, so the motor will turn clockwise.

3. Answer: A. The arrow on the motor block is pointing to the right, so the motor will turn clockwise.

4. Answer: B. The arrow on the motor block is pointing to the right, but a negative rotation value makes the motor turn in the opposite direction, so the motor will turn anti-clockwise.

5. Answer: C. The motor will not move because seconds cannot be negative number.

6. Answer: B. The arrow on the motor block is pointing to the right, but a negative speed value makes the motor turn in the opposite direction, so the motor will turn anti-clockwise.

7. Answer: A. The arrow on the motor block is pointing to the right, but a negative speed value makes the motor turn in the opposite direction, so the motor will turn anti-clockwise. But because the rotation value is negative, it cancels out the anticlockwise command, making the motor spin clockwise again. Hence clockwise is the correct answer.

8. Answer: C. The motor will not move because seconds cannot be negative number.

9. Answer: B. The arrow on the motor block is pointing to the right, but a negative speed value makes the motor turn in the opposite direction, so the motor will turn anti-clockwise.

10. Answer: A. When someone is closer than 90 cm, the condition becomes true. So the motor block runs and the arrow on the block points to the right, which means the motor will turn clockwise.

11. Answer: D. Someone more than 120 cm away is not less than 100 cm, the condition is false, so the motor does not turn and the robot shows the words "start game" instead.

12. Answer: B. When someone is closer than 90 cm, the condition becomes true. So the motor block runs and the arrow on the block points to the right, which means the motor will turn clockwise. But a negative speed value makes the motor turn in the opposite direction, so the motor will turn anti-clockwise

13. Answer: D. Someone more than 120 cm away is not less than 100 cm, the condition is false, so the motor does not turn and the robot shows the words "start game" instead.

14. Answer: D. Someone 100 cm away is not less than 100 cm, the condition is false, so the motor does not turn and the robot shows the words "start game" instead.

15. Answer: D. The robot writes "Start game" only when the "if" condition is false. The condition checks whether the distance is less than 100 cm, so your hand must be more than 100 cm away, which matches option D (150 cm away).

Range and Conditionals with Ultrasonic Sensor

Coding Concepts

1. Answer: D. If the ultrasonic sensor measures a very short distance, it means there's something very close to it.

2. Answer: B. Ultrasonic sensor uses sound waves to detect an obstacle.

3. Answer: B. When an object is placed closer to the ultrasonic sensor, the distance reading becomes smaller

4. Answer: B. the ultrasonic sensor tells the robot how close an object is, we can use that information to program the robot to react whenever something is nearby.

5. Answer: B. If the ultrasonic sensor measures a very long distance, it means the object is far from the sensor.

6. Answer B. If ultrasonic sensor measures a short distance, the robot can use that information to avoid bumping into the object. For example, If the robot senses a wall 5 cm in front of it, it can stop moving or turn left/right to avoid crashing into the wall.

7. Answer: A. Base on the code, the robot moves faster only when the ultrasonic sensor senses a distance less than 100 cm. Among the options, 80 cm is the only distance below 100 cm

8. Answer: D. Base on the code, the robot moves slower only when the ultrasonic sensor senses a distance not less than 100 cm. Among the options, 110 cm is the only distance above 100 cm.

9. Answer: B. Base on the code, the robot moves faster only when the ultrasonic sensor senses a distance is equal to 100 cm. Hence B is the correct answer.

10. Answer: B. Because 150 cm is not equal to 100 cm, the condition in the 'if' condition is false, so the robot follows the 'else' instructions and moves at the slower speed of 10%.

11. Answer: D. Based on the code, the robot moves anti-clockwise when the *'else'* block runs, which happens when the ultrasonic sensor does not sense a distance less than 100 cm. Among the options, only 120cm is not less than 100cm, so D is the correct answer.

12. Answer: A. Based on the code, the robot moves clockwise when the 'if' condition is true, which happens when the ultrasonic sensor senses a distance less than 100 cm. Among the options, only 99cm is less than 100cm, so A is the correct answer.

13. Answer: A. The robot should move fast only when the distance sensed is more than 15 cm, and option A uses the correct operator "> 15" to check for that condition.

14. Answer: B. The robot should move fast only when the distance sensed is less than 15 cm, and option B uses the correct operator "< 15" to check for that condition.

15. Answer: C. The robot should move fast only when the distance sensed is at exactly 15 cm, and option C uses the correct operator "= 15" to check for that condition.

Conditionals with Color Sensor

Coding Concepts

1. Answer: B. The color sensor in LEGO Spike is designed to detect different colors and shades, allowing the robot to react based on what color it sees.

2. Answer: B. The color sensor works by shining light onto an object and measuring the color of the light that bounces back. This is how it can identify different colors.

3. Answer: C. LEGO Spike color sensor uses visible light (red, green, and blue LEDs) to shine on an object and detect the color that reflects back.

4. Answer: D. The color sensor detects the color reflected from the object in front of it. If the object is red, it reflects red light back to the sensor, so the sensor will correctly identify and report red.

5. Answer: A. The color sensor can only detect one color at a time. It does not scan multiple colors simultaneously.

6. Answer: B. This code shows one if block is inside another if block, making it a nested ifs, not multiple ifs. Multiple-ifs means several separate "if" statements at the same level..

7. Answer: A. The code shows two separate "if" blocks placed at the same level, one after another.

8. Answer: A. The code still contains more than one "if" condition being checked, even though one of them is inside an else block. This structure is called if–else if, which is also considered a multiple-ifs condition.

9. Answer: C. The code uses two separate "if" statements, both checking the same condition, whether the color sensor detects red. Since multiple independent if-statements are used (not if-else), both conditions will be true at the same time, and the robot may try to run both sounds, causing it to play either Crazy Laugh or Boop Bing Bop.

10. Answer: C. The code only checks for red and blue. Since green is not one of the colors in the code, neither of the "if" statements becomes true, so the robot has no instructions to follow when it senses green, meaning it will simply do nothing.

11. Answer: C. The code only checks for red and blue. Since purple is not one of the colors in the code, neither of the "if" statements becomes true, so the robot has no instructions to follow when it senses purple, meaning it will simply do nothing.

12. Answer: C. The first "if" block checks whether the color sensor detects red, and play the "Crazy Laugh" sound.
13. Answer: B. The robot always checks the red condition first, since that if-block is placed above the blue one. When it senses blue, the red condition is false, so the robot will not do anything.

14. Answer: B. The motor block is inside the second "if" statement, which only runs when the color sensor detects red. This means the Spike Spinning Top will spin only when the sensor senses red, not yellow or any other color.

15. Answer: A. The Crazy Laugh sound block is inside the first “if” statement, which only runs when the color sensor detects yellow.

More Conditionals with Color Sensor

Coding Concepts

1. Answer: B. A color sensor's main purpose in LEGO Spike robotics is to detect different colors, allowing the robot to respond with actions based on the color it senses.

2. Answer: B. The color sensor tells the robot what color it sees, and you can use that information in your code to make the robot react.

3. Answer: B. A color sensor can detect the difference between light and dark surfaces, allowing the robot to see and follow a black line on the floor.

4. Answer: C. When a robot detects a specific color, such as yellow, you can program it to perform any action you choose, such as turning right. The other options are unrealistic or impossible for a LEGO robot.

5. Answer: C. If the color sensor detects a color that is not the one you programmed a reaction for, the robot will not match any condition in the code. Since no instructions are given for that color, the robot simply does nothing.

6. Answer: C. When the color sensor detects a color you programmed it to recognize, it sends a signal to the robot, telling it that the condition is met. This triggers whatever action you coded.

7. Answer: A. You can program the robot to perform real, achievable actions, such as turning left when it detects a certain color. The other options are unrealistic or impossible for a LEGO robot.

8. Answer: D. A SPIKE color sensor can detect real, solid colors like green. The other options are not actual single colors, and brown is not typically included in the standard set of colors the sensor can detect.

9. Answer: B. Only the color sensor can detect specific colors like red. If you want the robot to stop when it detects red, the robot needs a sensor that can recognize color, which the ultrasonic, touch, and gyroscope sensors cannot do.

10. Answer: D. The color sensor works by shining light onto an object and measuring the color of the light that bounces back. This reflected light tells the sensor what color it is seeing.

11. Answer: C. The primary function of a LEGO Spike color sensor is to detect colors and let the robot react based on those colors.

12. Answer: B. The forever block is missing, so the code runs only once when the program starts. Since the program does not loop continuously, the action is not guaranteed.
13. Answer: C. The forever block is missing, so the code runs only once when the program starts. Since the program does not loop continuously, the action is not guaranteed.

14. Answer: Left. The left code is better because the forever loop keeps checking the color sensor continuously, so the Spike Hopper will react immediately whenever it sees red or blue. The right code only checks the color once at the start, so it won't respond again after that.

15. Answer: Right. The right code is better because the wait block is at the bottom of the loop, which lets the sound finish playing for 1 second before the next color check. This makes the robot respond smoothly without cutting off the sounds.