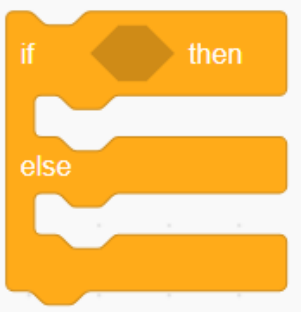
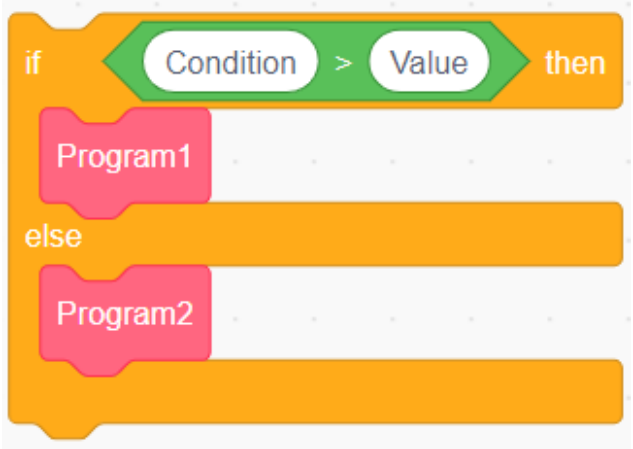
Course Review

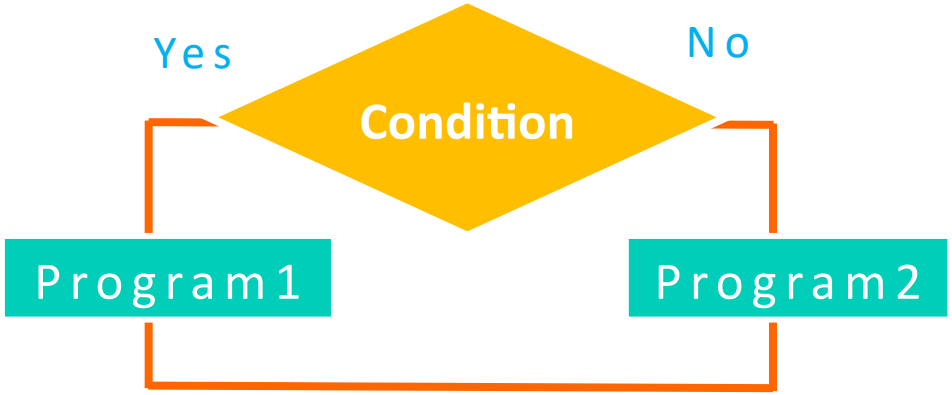
1. Branching structure

Branching structure: Choose different execution paths based on different conditions.

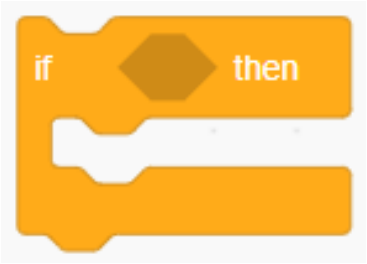
The "If-Then-Else" module can implement different program selections based on conditional statements.



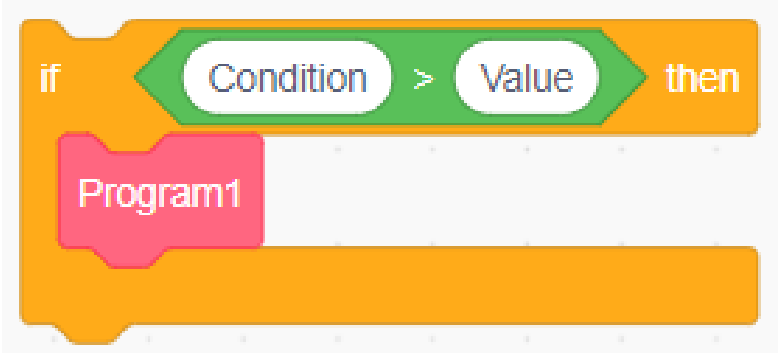


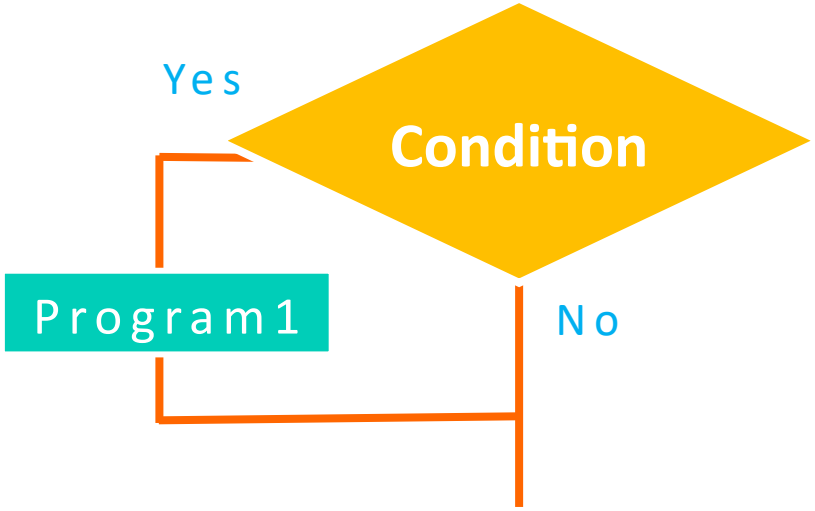


Course Review



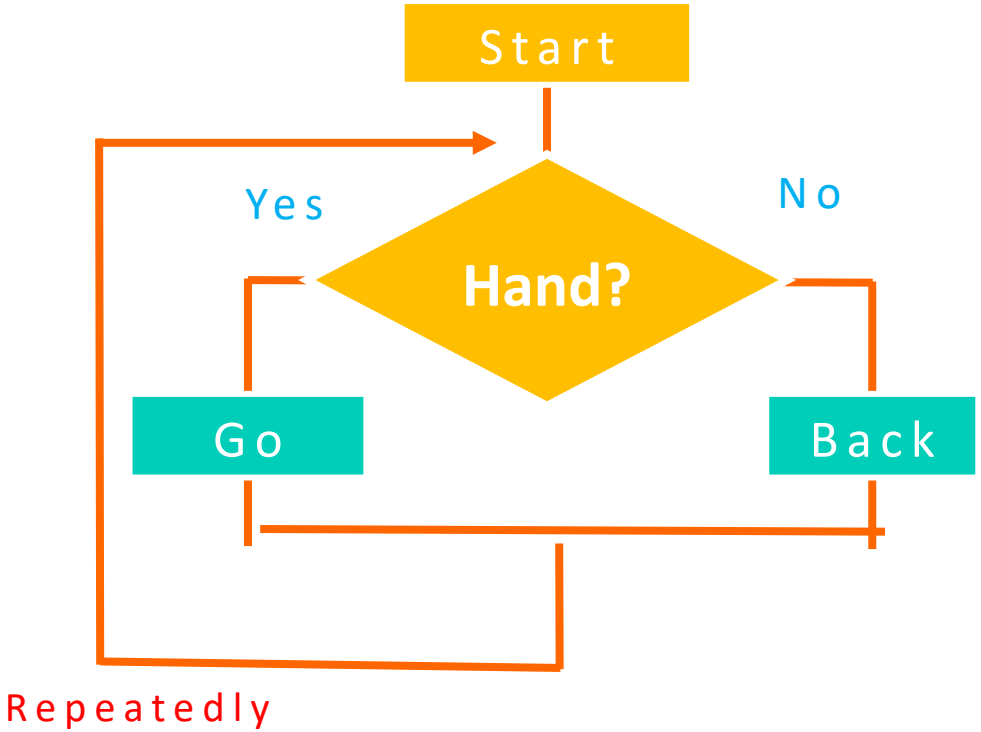
The "If-Then" model can implement actions that execute the program if certain conditions are met, and do not execute if they are not met.

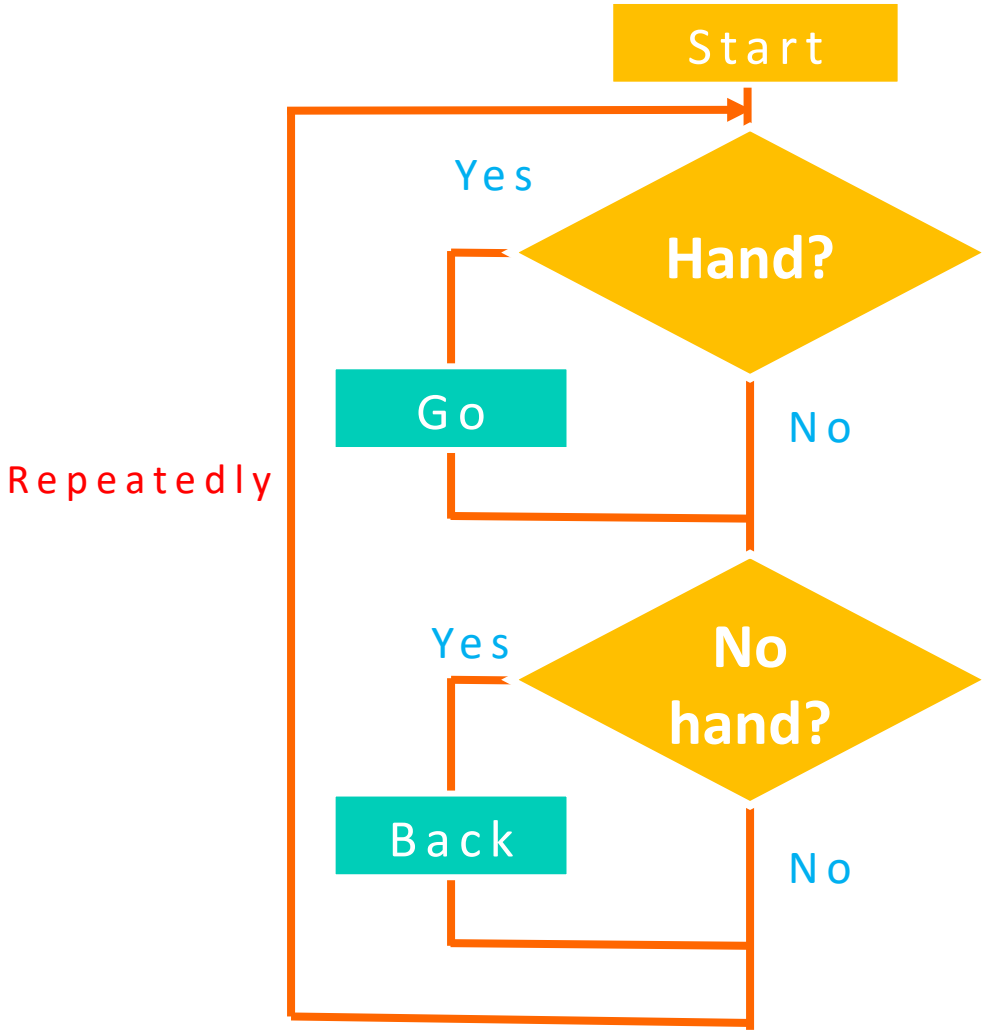




Course Review

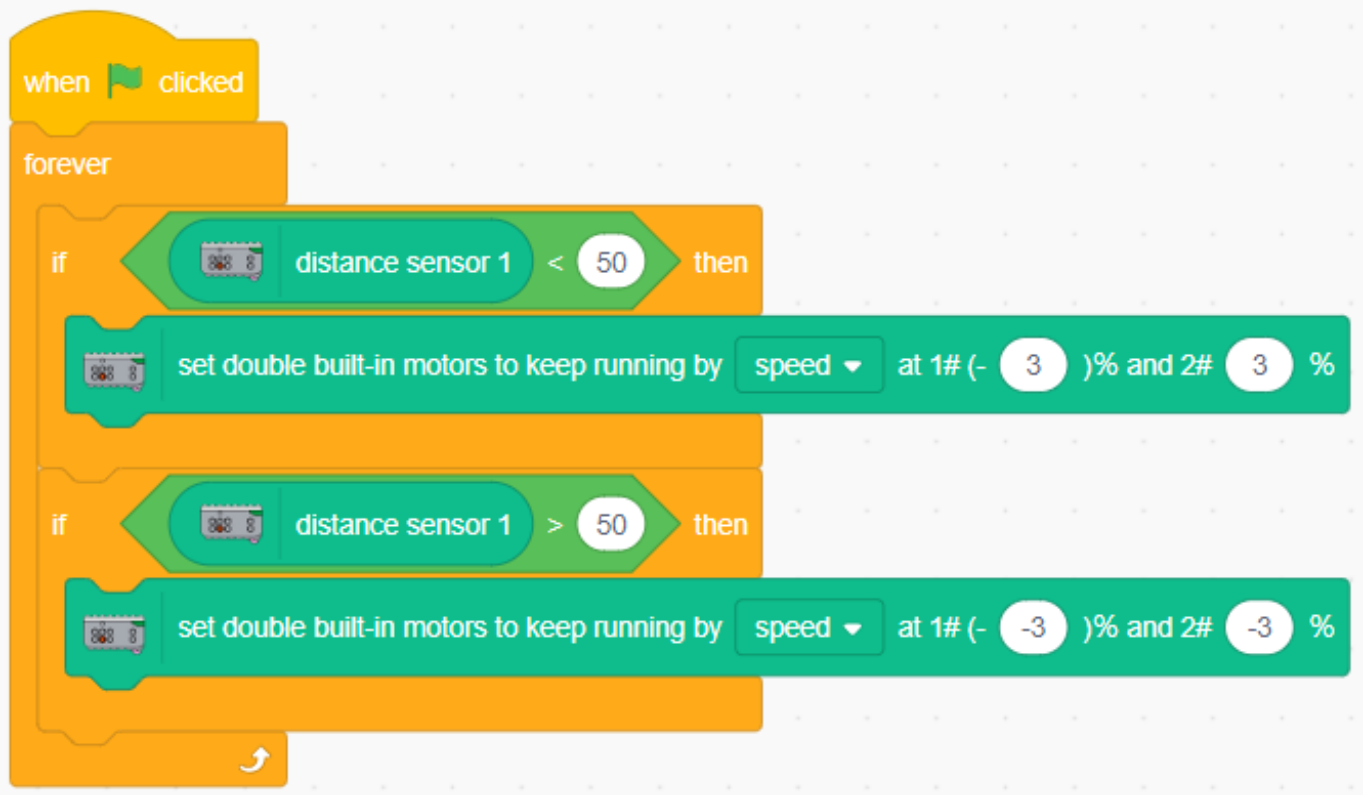
2. Program Explanation

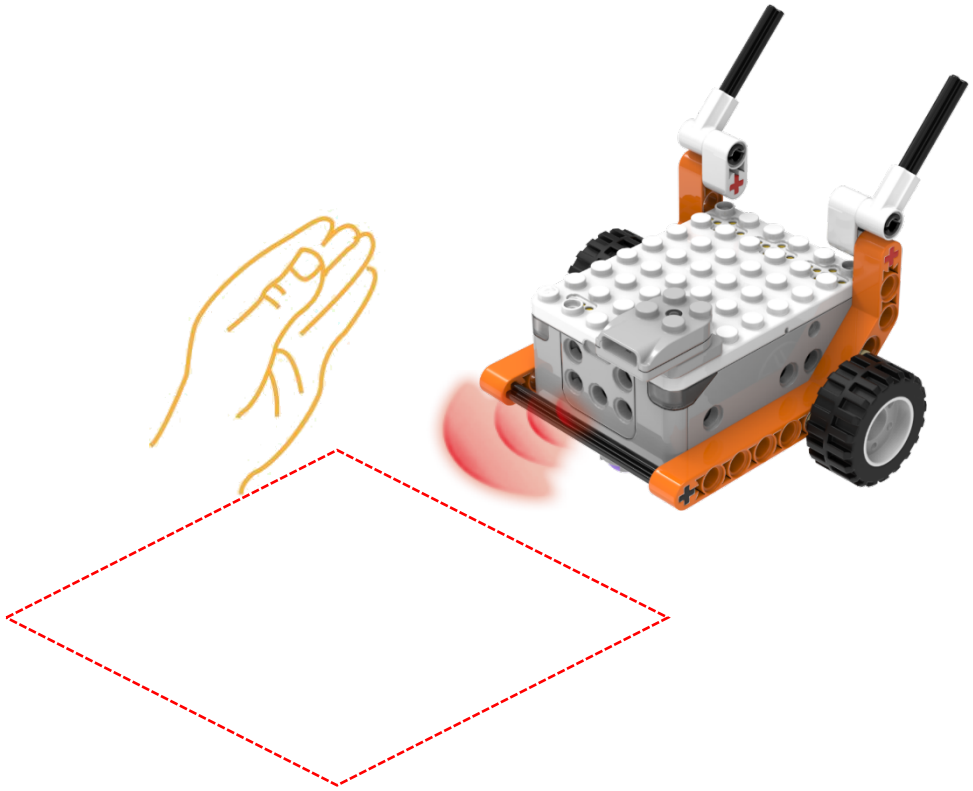




Course Review

3. Finish the task





Please use your own kit.

Do not put any parts in your mouth.

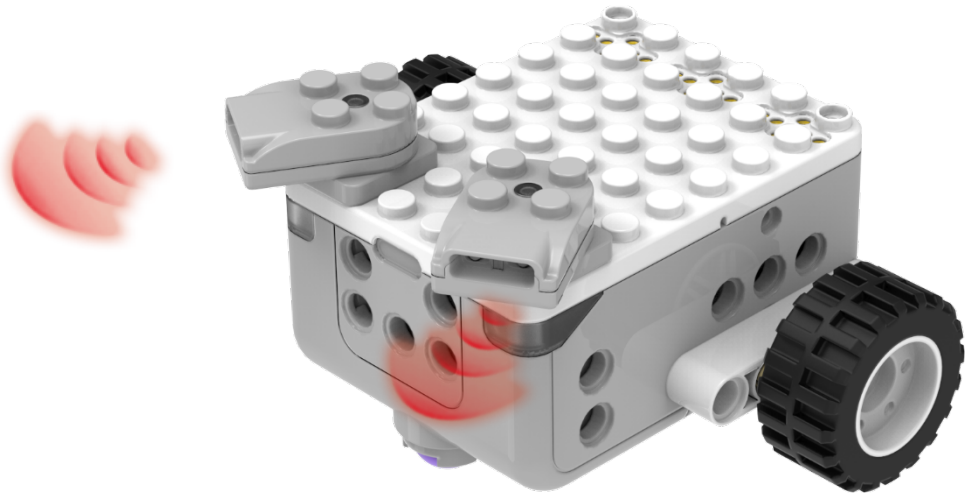
Please clean up after use.

Please raise your hand if you have any questions.

INTRODUCTION

Scenarios

In the last lesson, we implemented gesture control for the car to move forward and backward (and stop). However, a regular racing car has four functions: moving forward, turning left, turning right, and stopping. We can program the car to incorporate these four functions.



Kids:

How many sensors are needed to achieve the four functions?

How can we implement four different actions through programming?

Let’s embark on our exploration journey with "gesture-controlled racing"!

暗物智能
DARKMATTER AI

Gesture-Controlled Racing

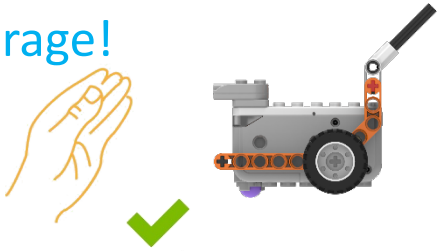
AI Courses

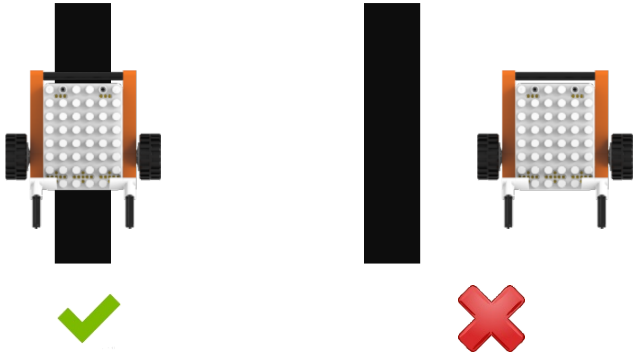
Scenarios

Competition rules:

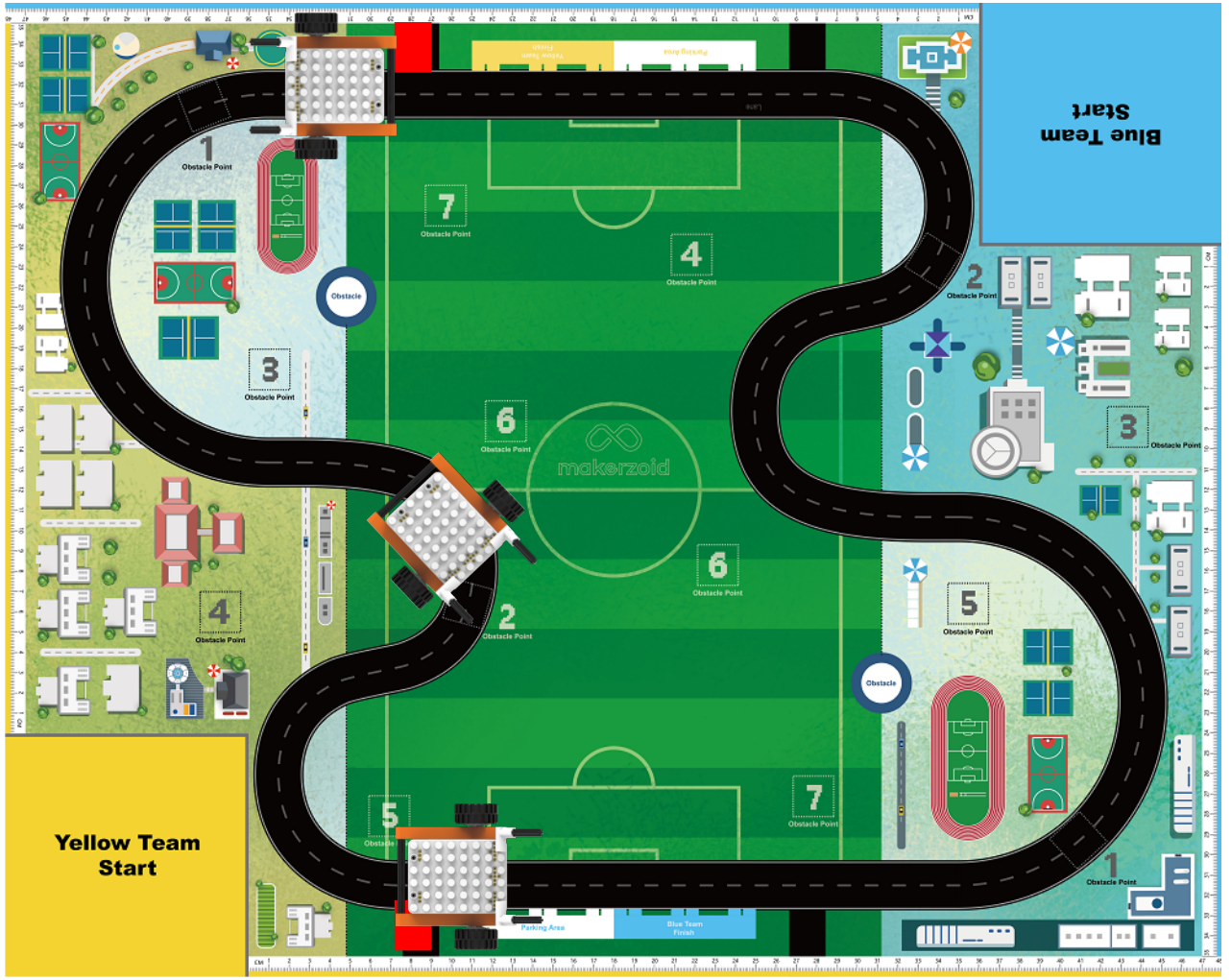
1. The contestant's car starts from the starting point and reaches the finish line through gesture control.

2. Let’s see who can control the car to stop in the garage!





Finish Point



Start Point

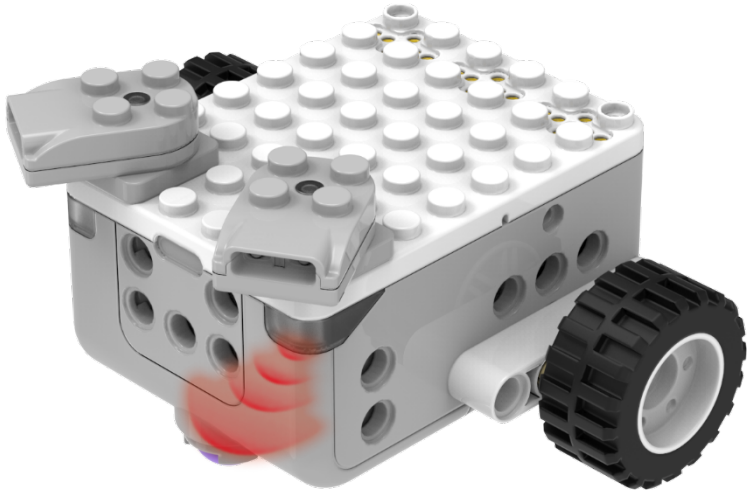
Scenarios

Question ：

Kids, do you know:

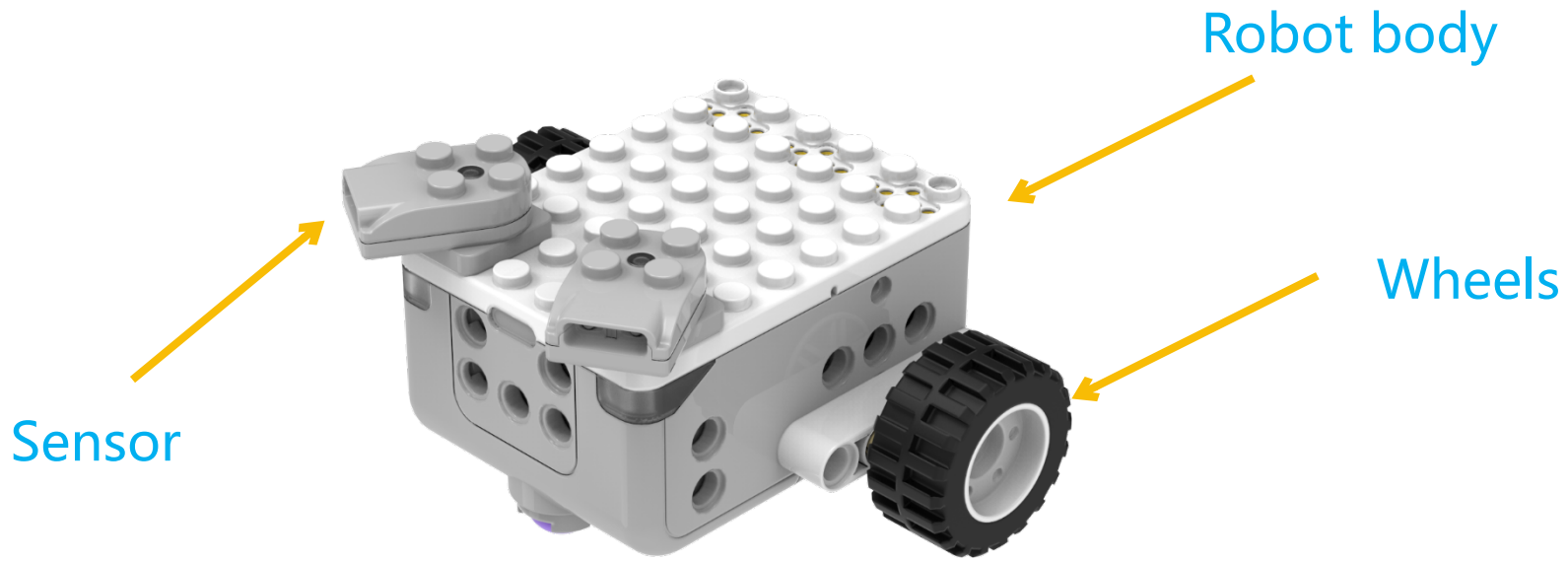
How many sensors are needed to achieve the four functions?

How can we implement four different actions through programming?



Scenarios

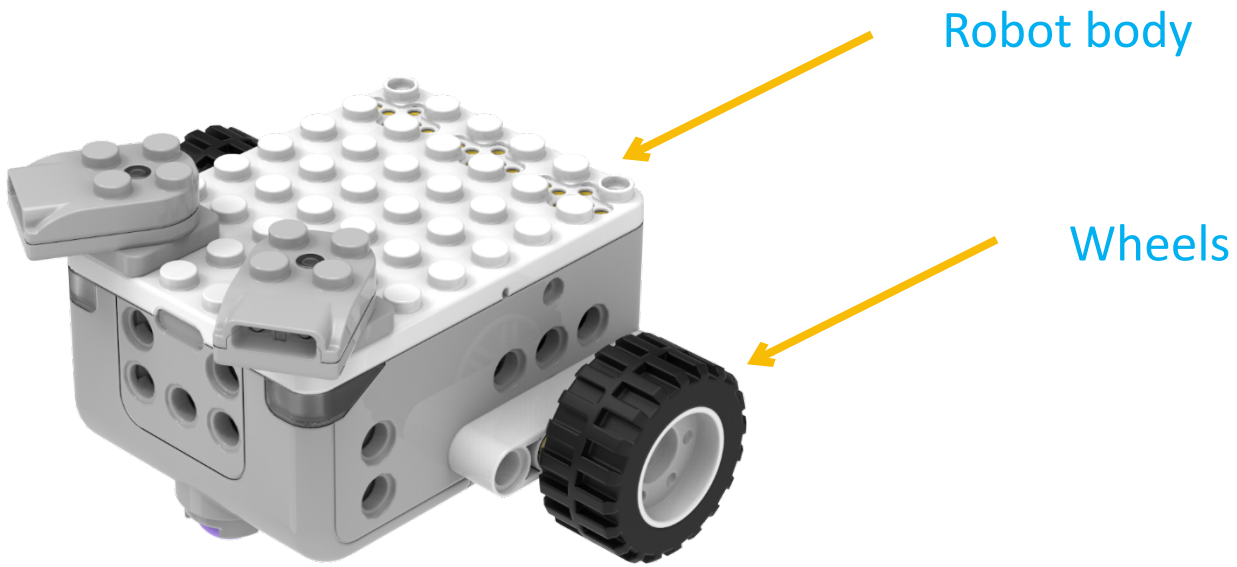
Today, everyone is a junior engineer. Let's work together to complete the gesture-controlled racing!



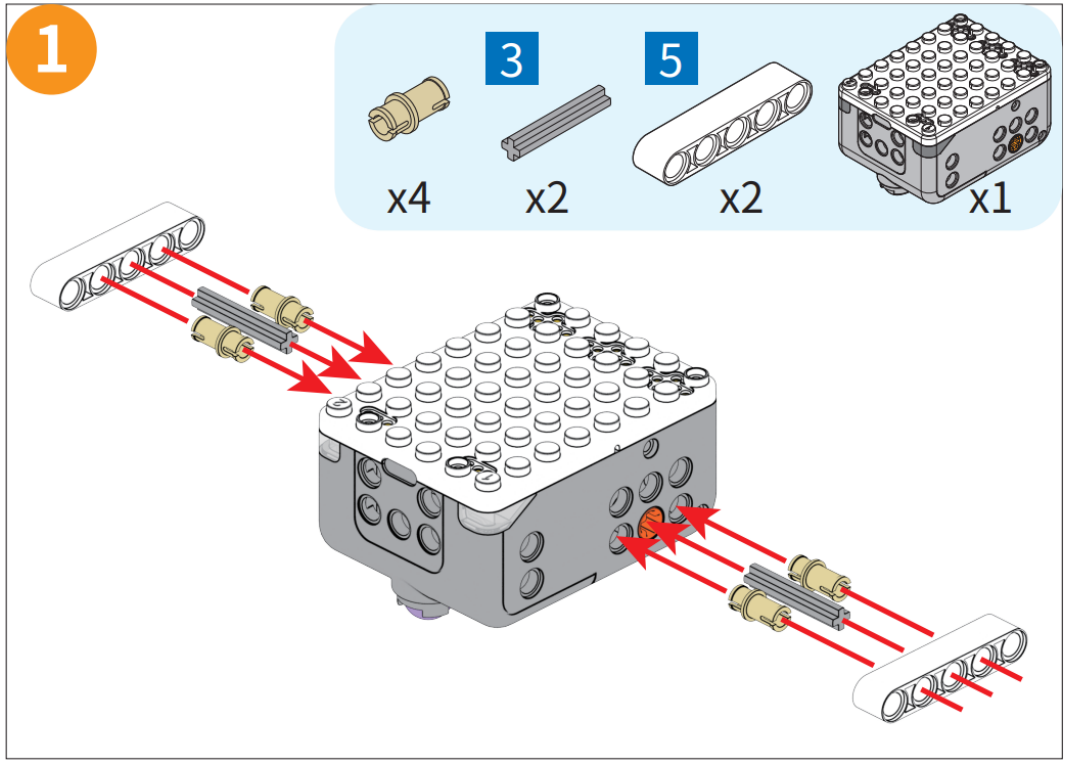
ASSEMBLY

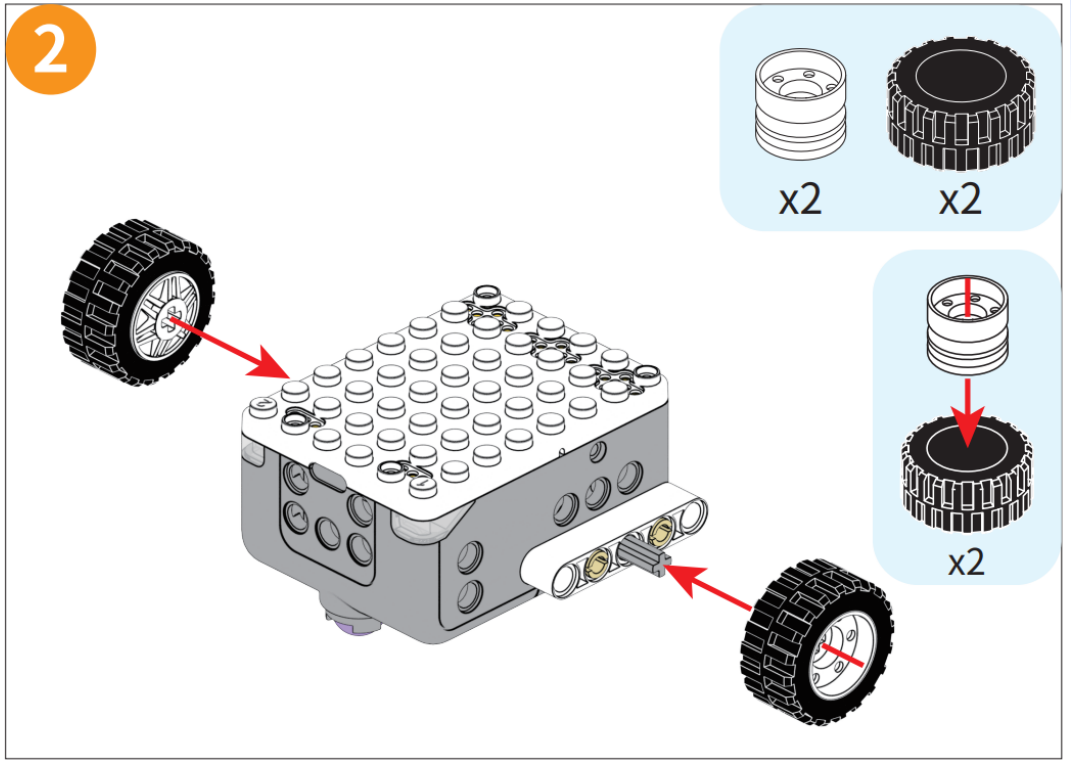
The Final Model

Robot body assembly



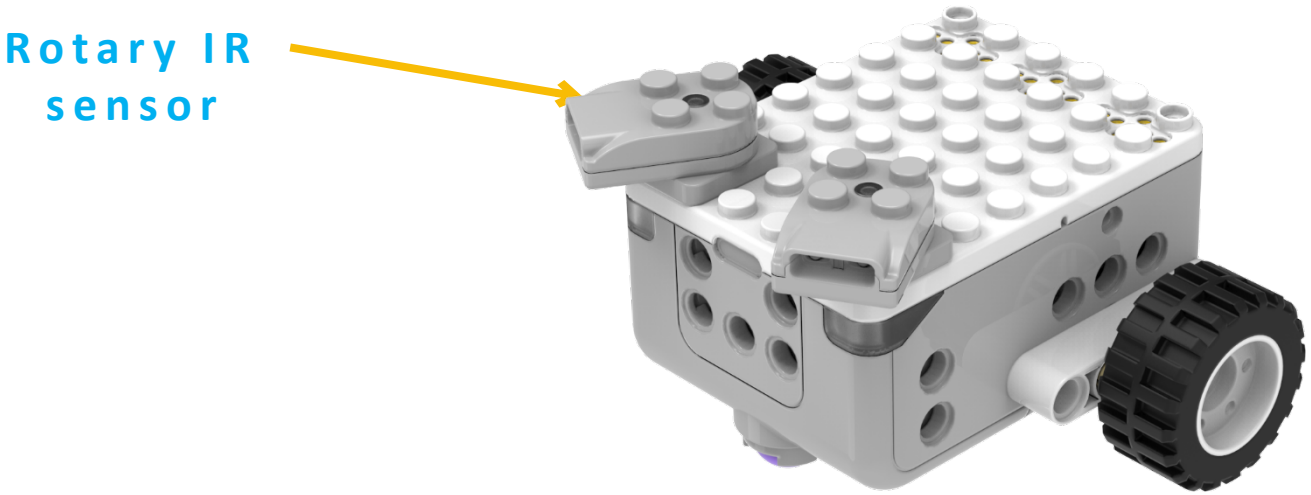
Assembly





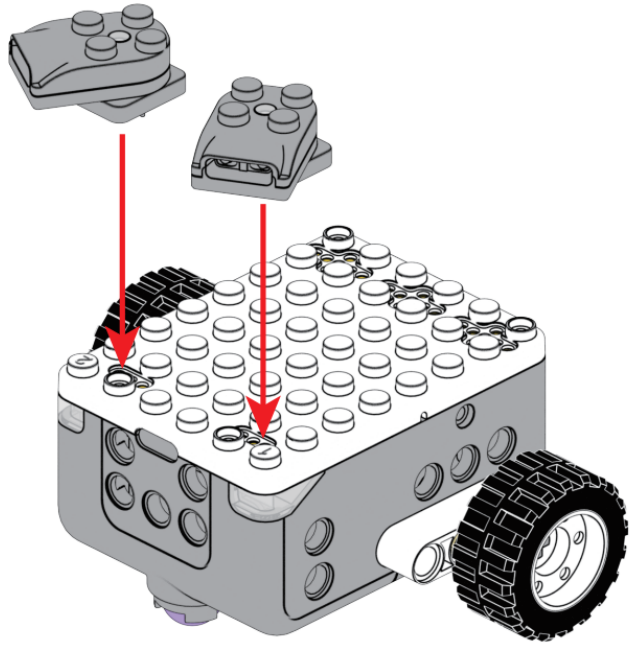
Assembly

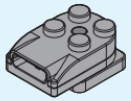
Assemble the rotary IR sensor



Assembly

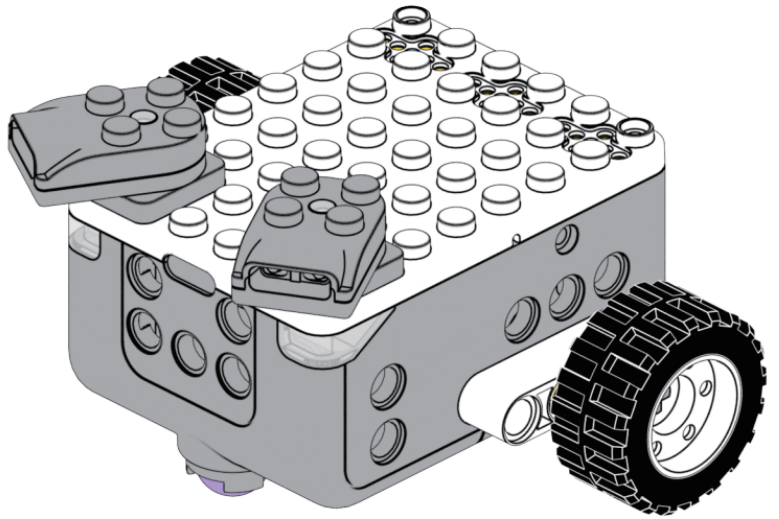
3





x2

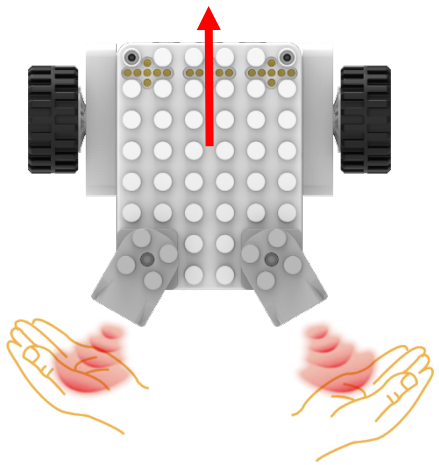
4



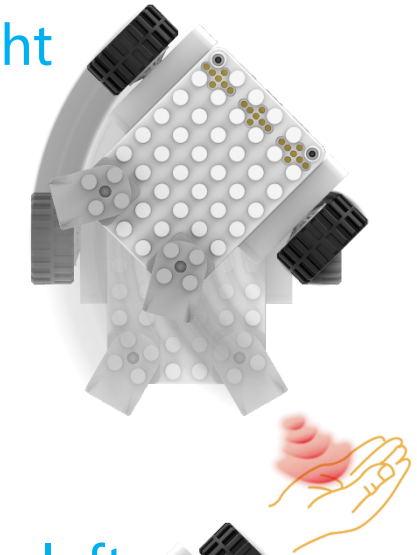
Assembly

Required Functions

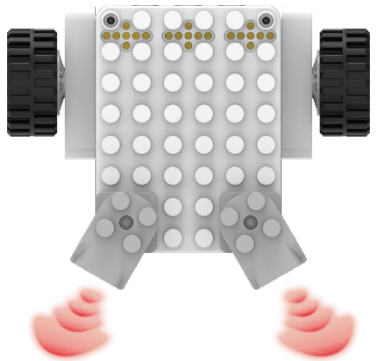
Go forward



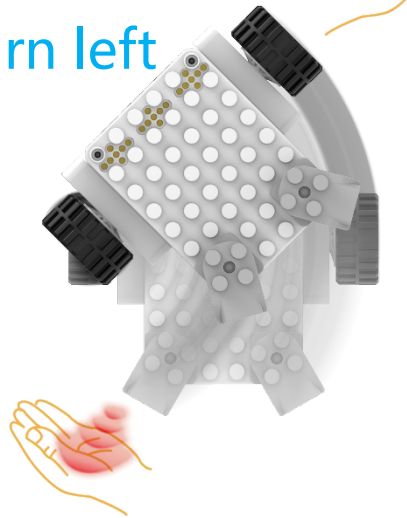
Turn right



Stop



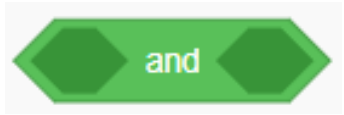
Turn left



PROGRAMMING

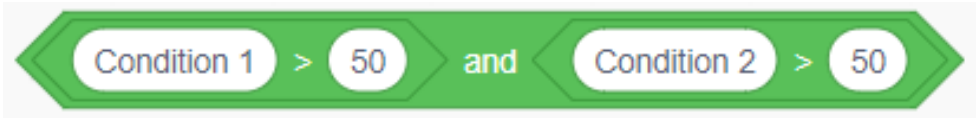
Introductions

Module Explanation



If both conditions are true, output "true"; otherwise, output "false."

For example :



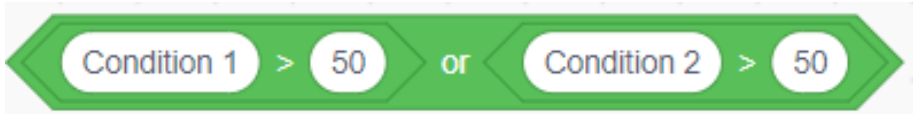
1. If condition 1 is 80 and condition 2 is 70, both conditions are met, so the output is "true."
2. If condition 1 is 30 and condition 2 is 70, with only one condition met, the output is "false."
3. If condition 1 is 70 and condition 2 is 30, with only one condition met, the output is "false."
4. If condition 1 is 30 and condition 2 is 30, with no conditions met, the output is "false."

Introductions

Module Explanation

If at least one condition is true, output "true"; if no conditions are true, output "false."

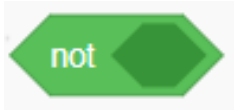
For example :



1. If condition 1 is 80 and condition 2 is 70, both conditions are met, so the output is "true."
2. If condition 1 is 30 and condition 2 is 70, with one condition met, the output is "true."
3. If condition 1 is 70 and condition 2 is 30, with one condition met, the output is "true."
4. If condition 1 is 30 and condition 2 is 30, with no conditions met, the output is "false."

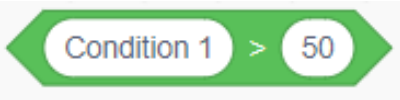
Introductions

Module Explanation



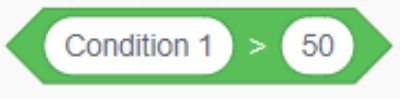
When the selected condition is not met, the output should be "true."

For example :

1. If condition 1 is 80, 

this condition is met.

the output is "false."

2. If condition 1 is 30, 

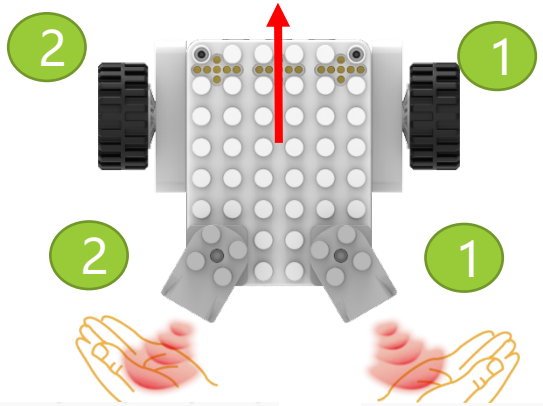
then this condition is not met.

the output is "true."

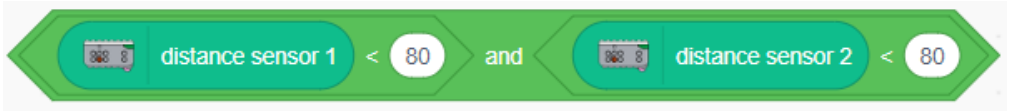
Introductions

Module Explanation

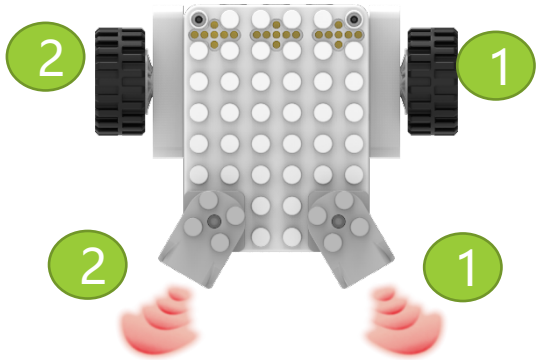
Go



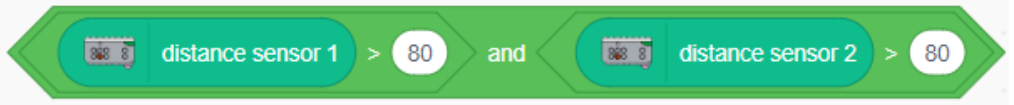
Two sensors detect simultaneously, causing the car to move forward. Therefore, we set the values of Sensor 1 and Sensor 2 to be less than 80 at the same time. If this condition is met, we output true.



Stop



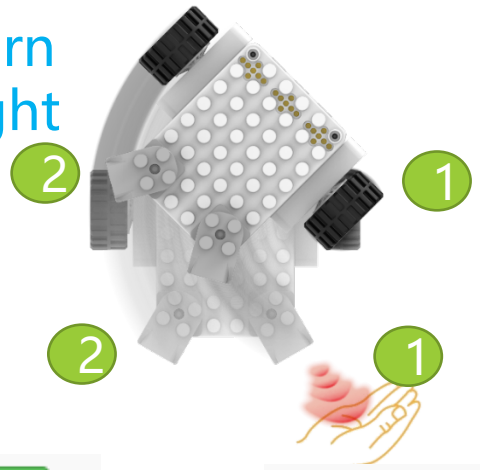
When both hands leave the sensors, the car stops. Therefore, we set the values of Sensor 1 and Sensor 2 to be greater than 80 at the same time. If this condition is met, we output true.



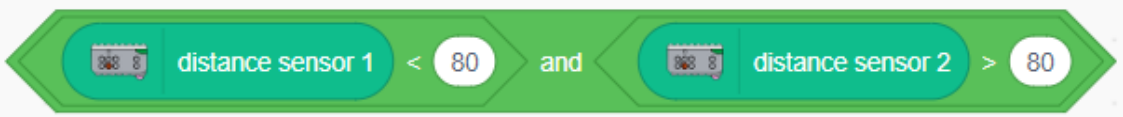
Introductions

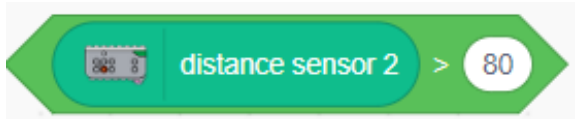
Module Explanation

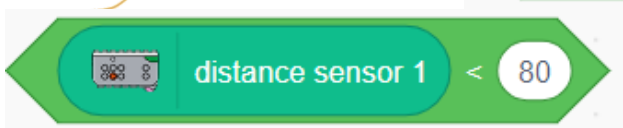
Turn right



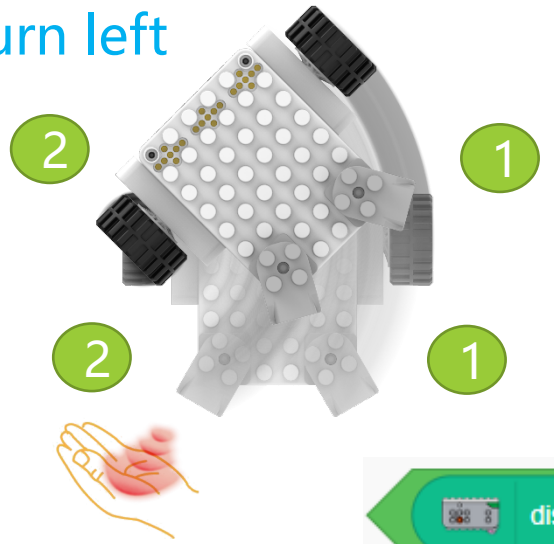
When Sensor 1 detects and Sensor 2 does not, the car moves to the right. Therefore, we set the value of Sensor 1 to be less than 80 and the value of Sensor 2 to be greater than 80. If this condition is met, we output true.



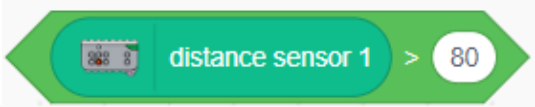


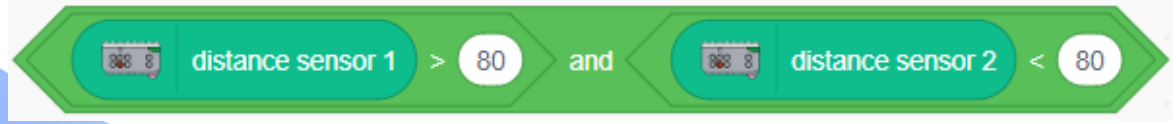


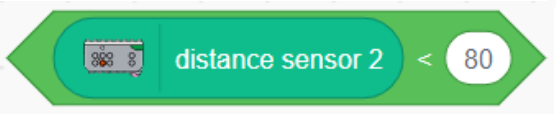
Turn left



When Sensor 1 does not detect and Sensor 2 does, the car moves to the left. Therefore, we set the value of Sensor 1 to be greater than 80 and the value of Sensor 2 to be less than 80. If this condition is met, we output true.

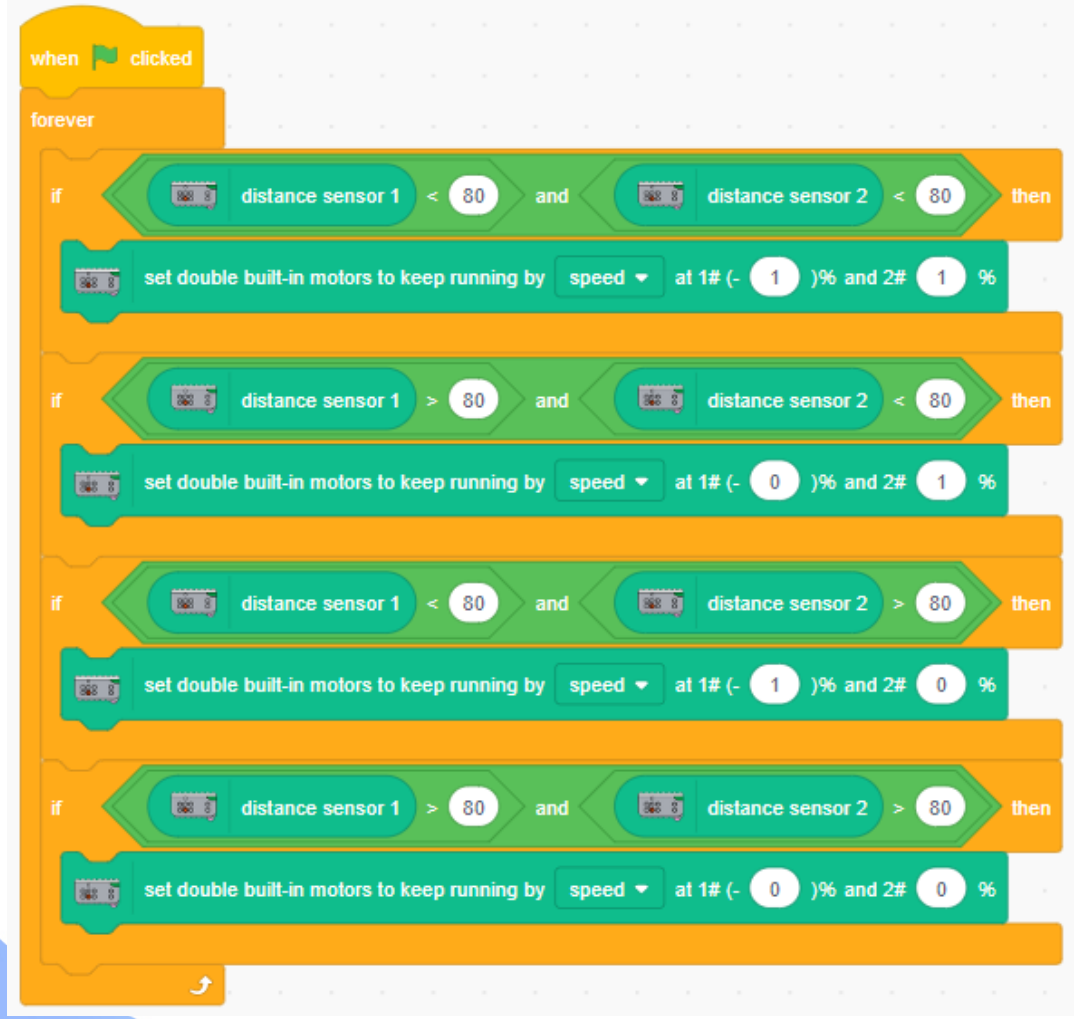


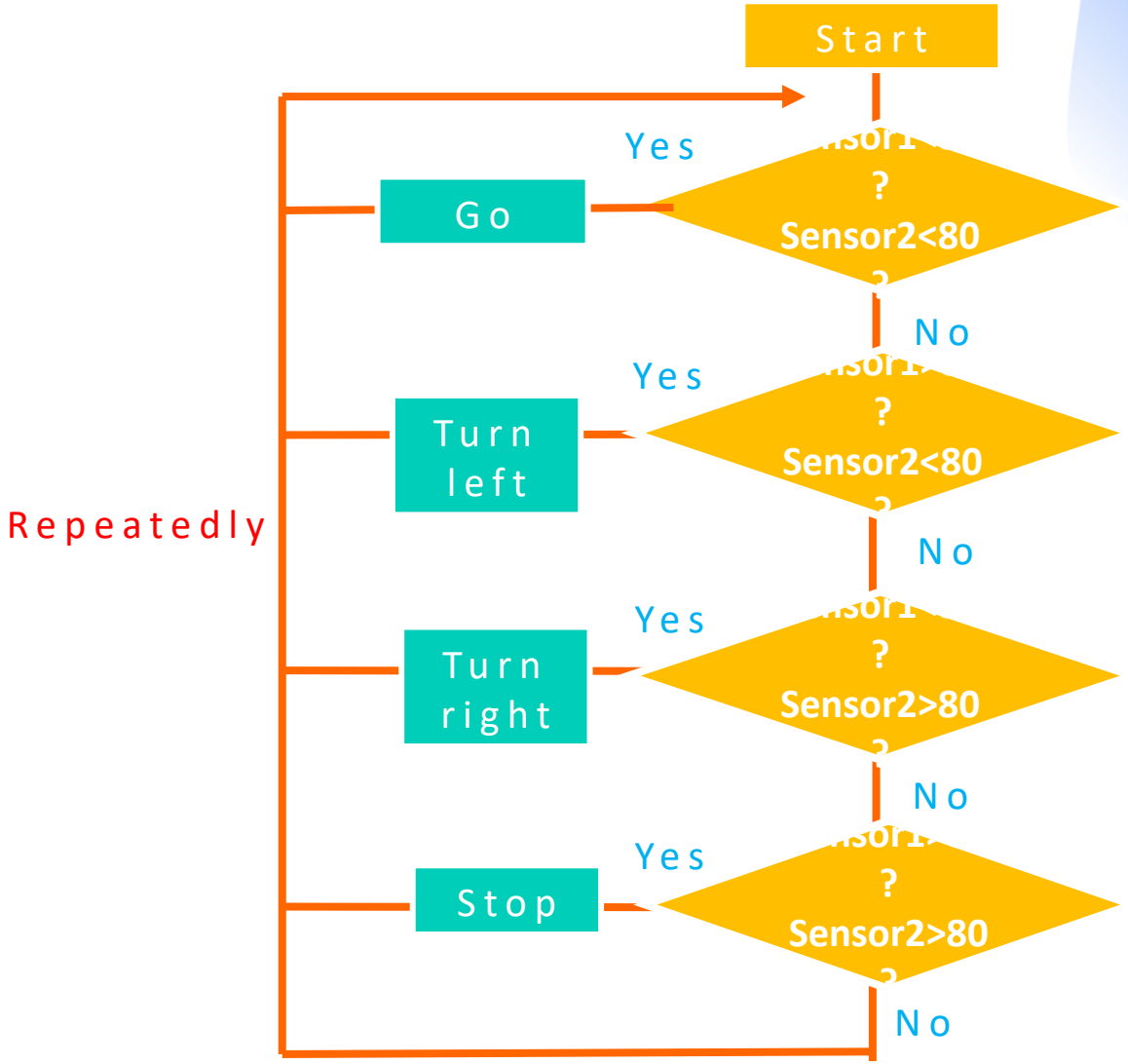




Introductions

Program Explanation

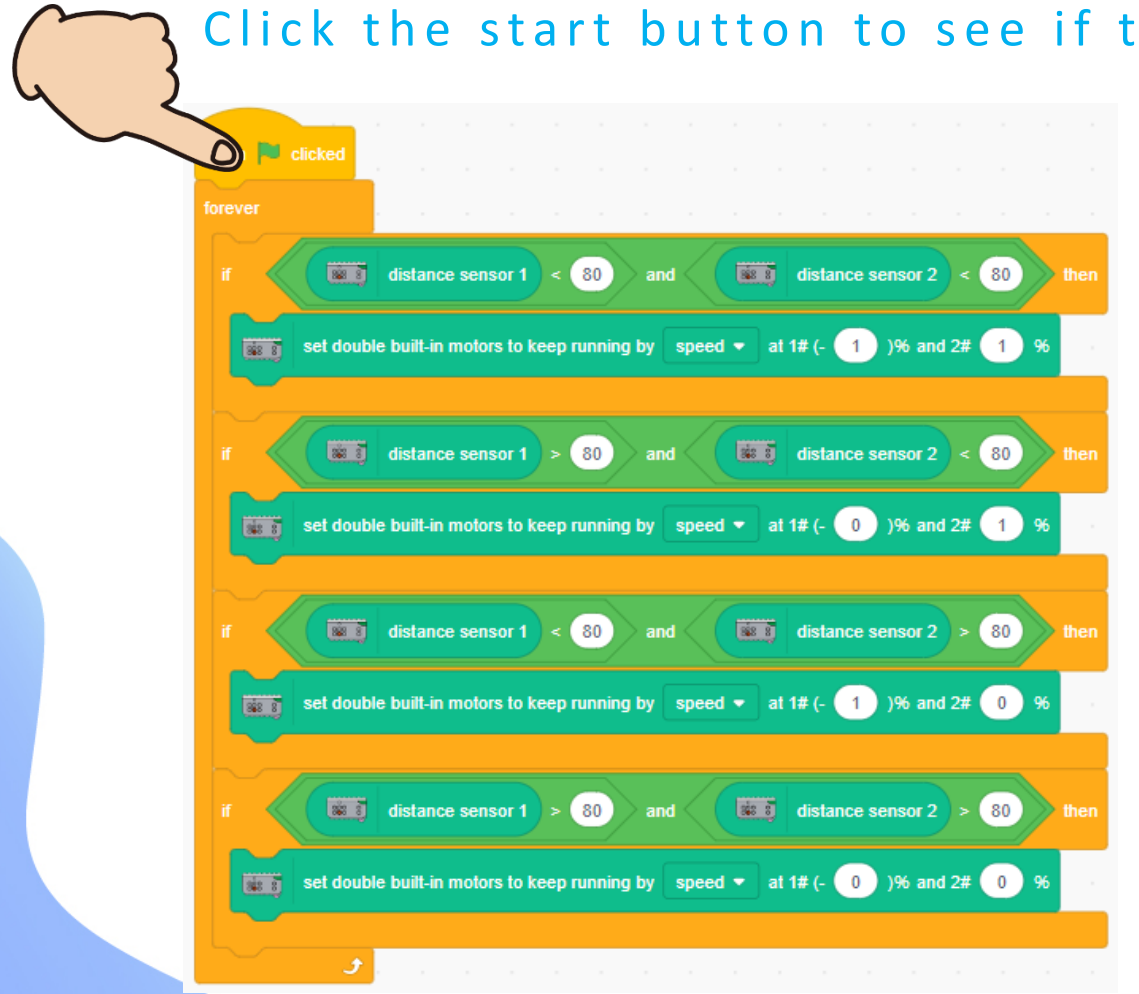


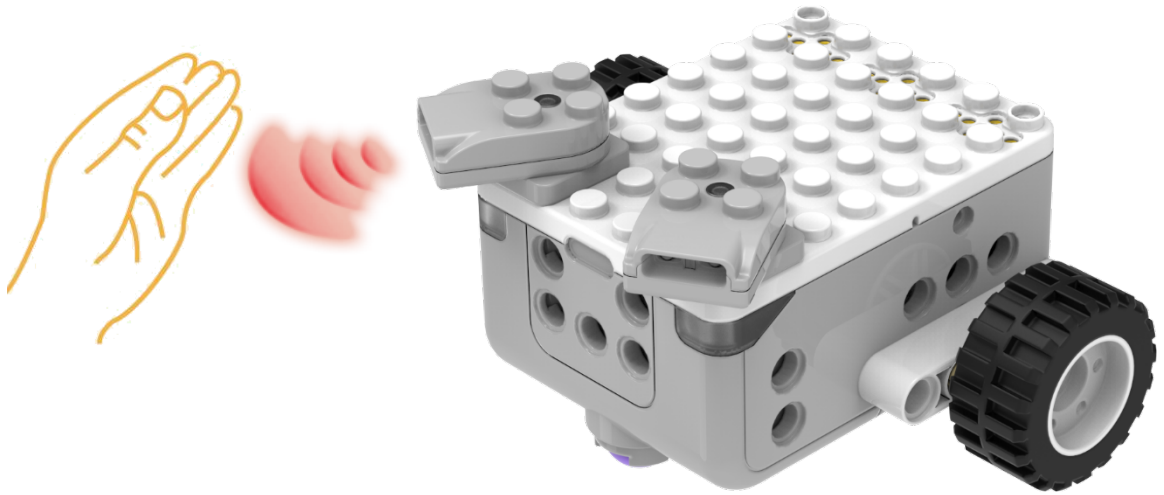


Introductions

Let’s play:

Click the start button to see if the robot can be controlled by gestures.

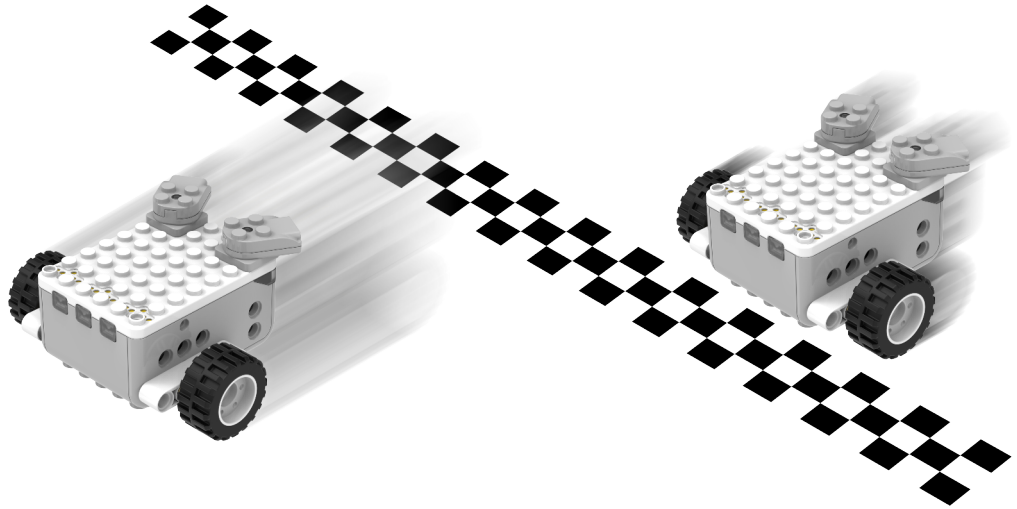




Play and Try

Let's compete:

The competitors are ready to see whose car reaches the finish line the fastest!

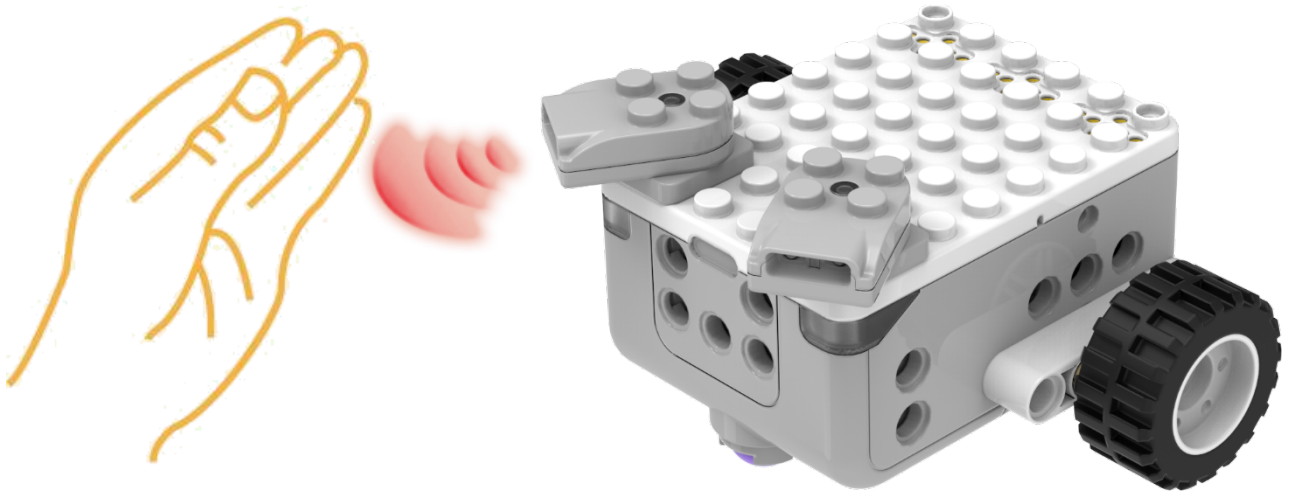


Kids, is there a way to make the robot even stronger?

CREATION

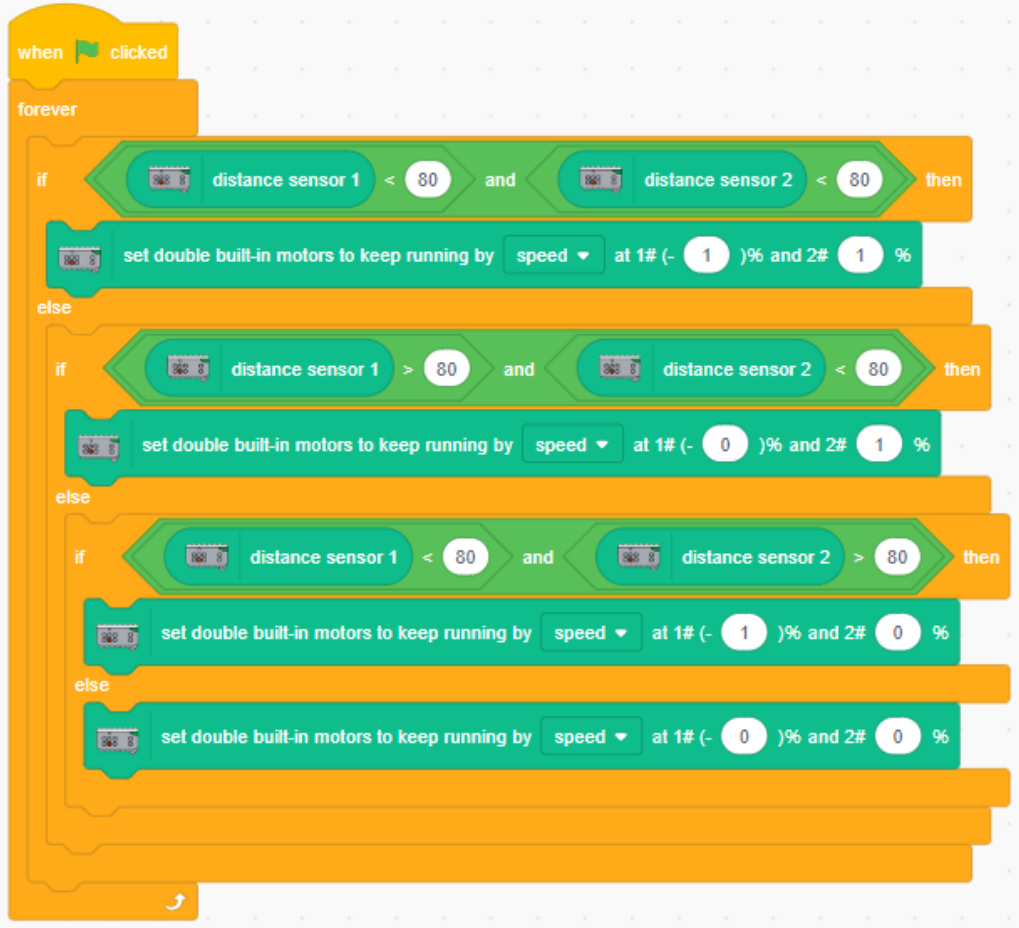
Create

1.Are there other ways to write the program?

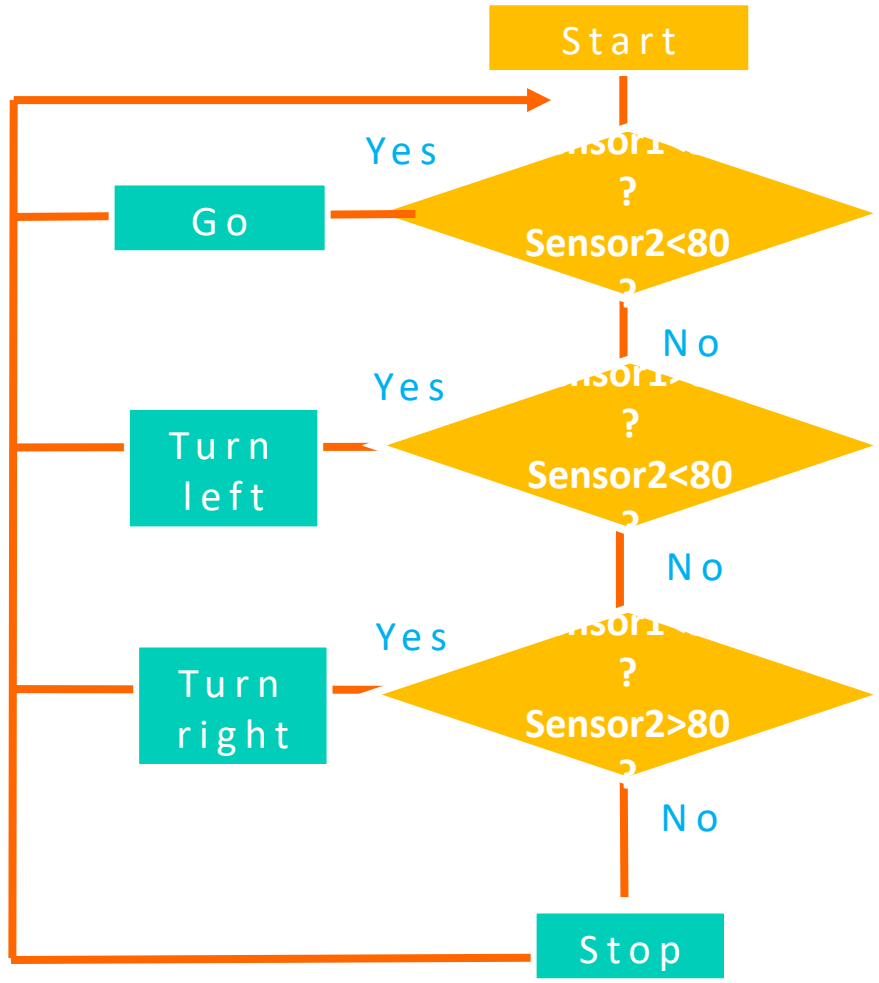


Introductions

Program Explanation



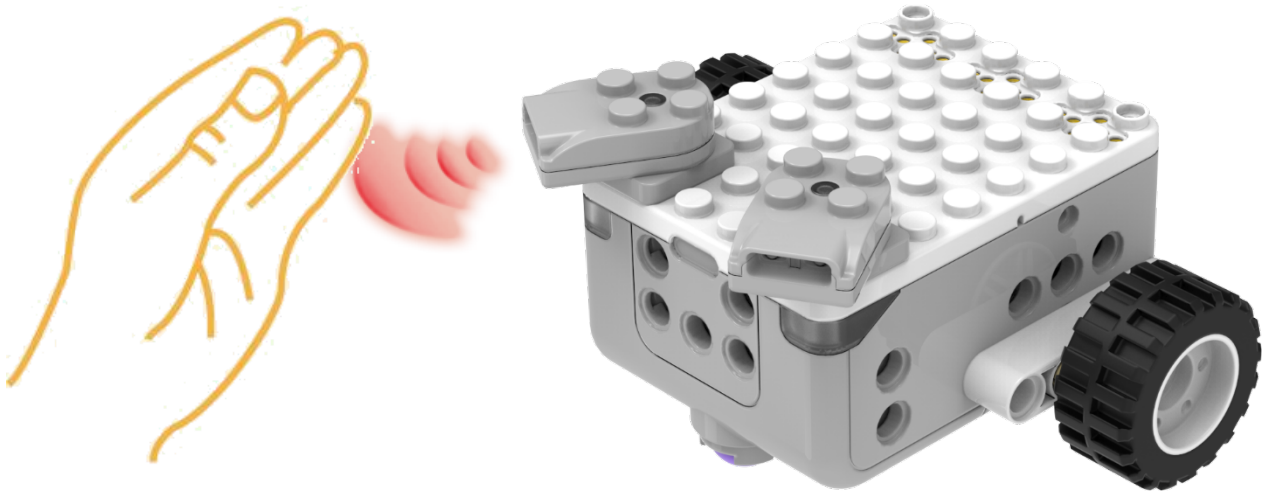
Repeatedly



Create

1.Are there other ways to write the program?

Use the new program to see if the car can run properly.

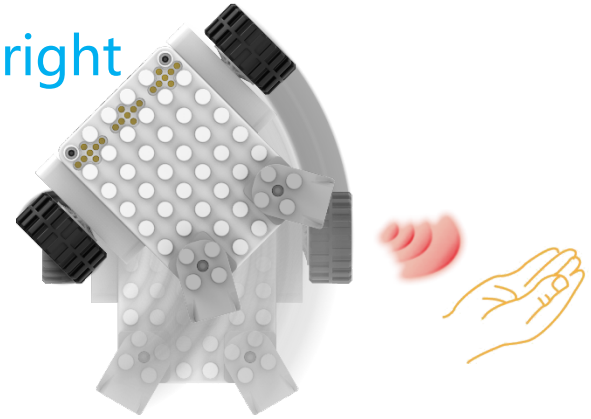


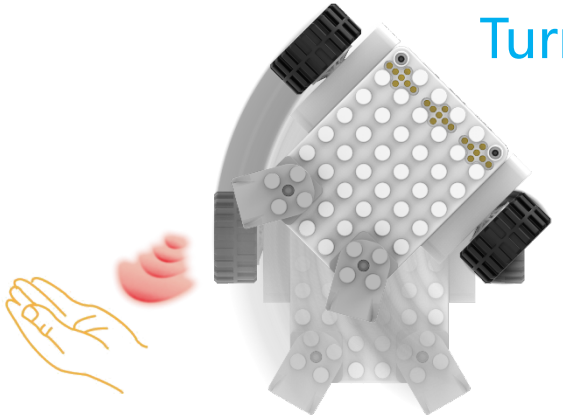
Create

2.Increase the difficulty of the competition.

Make sensor 2 control right turns and sensor 1 control left turns.

Turn right

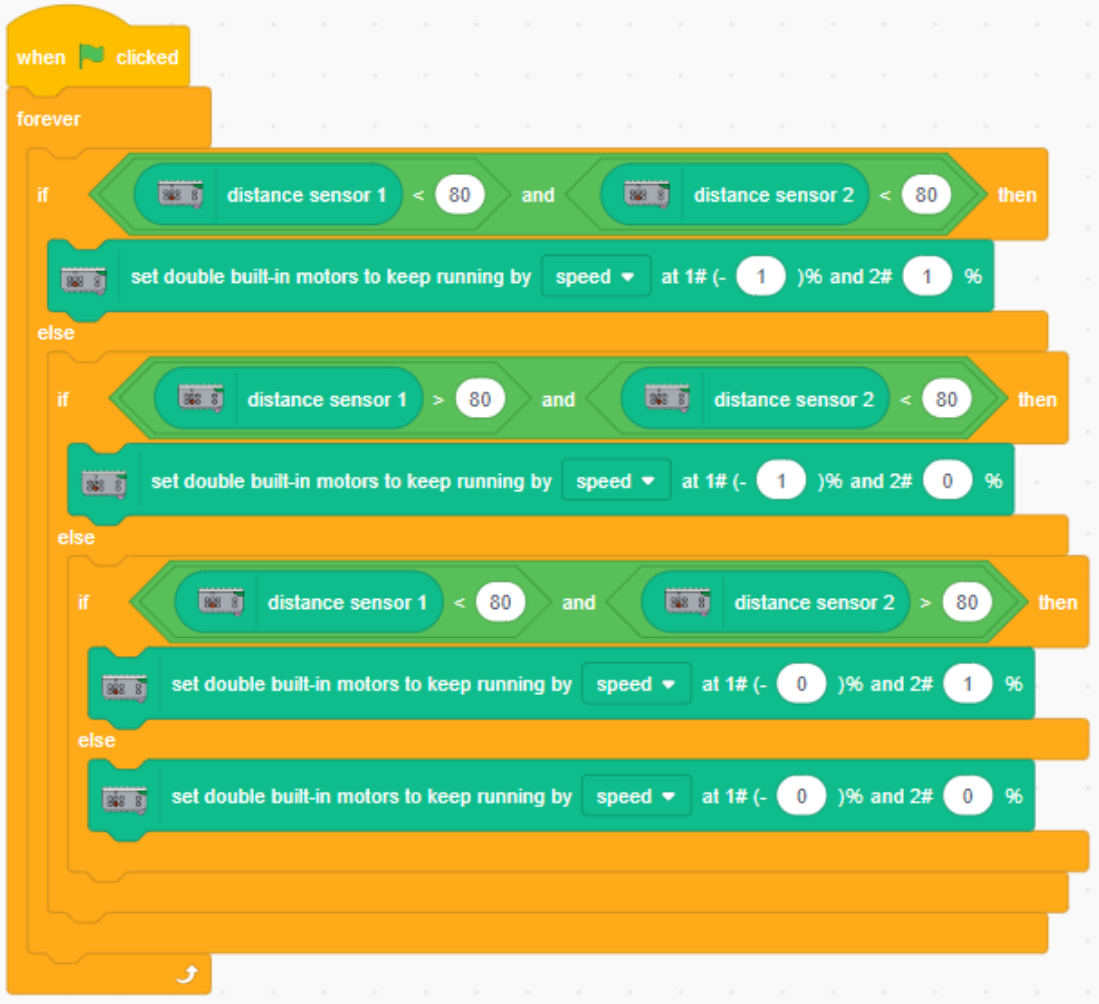




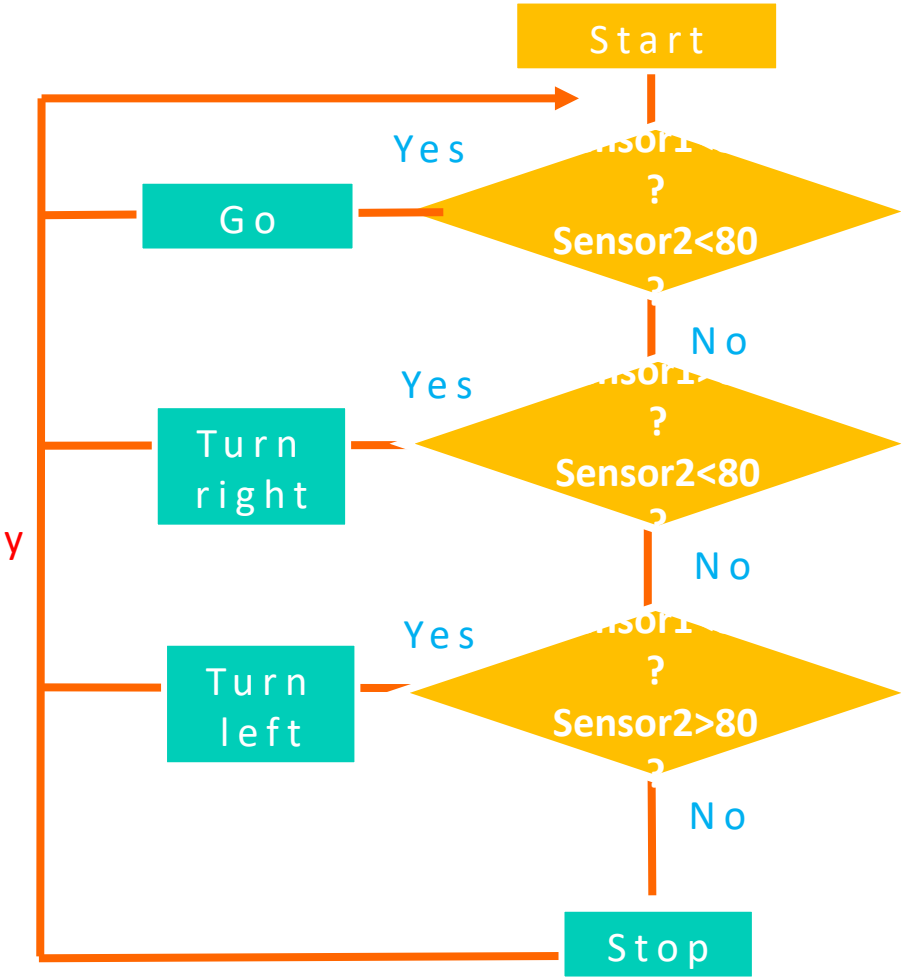
Turn left

Introductions

Program Explanation



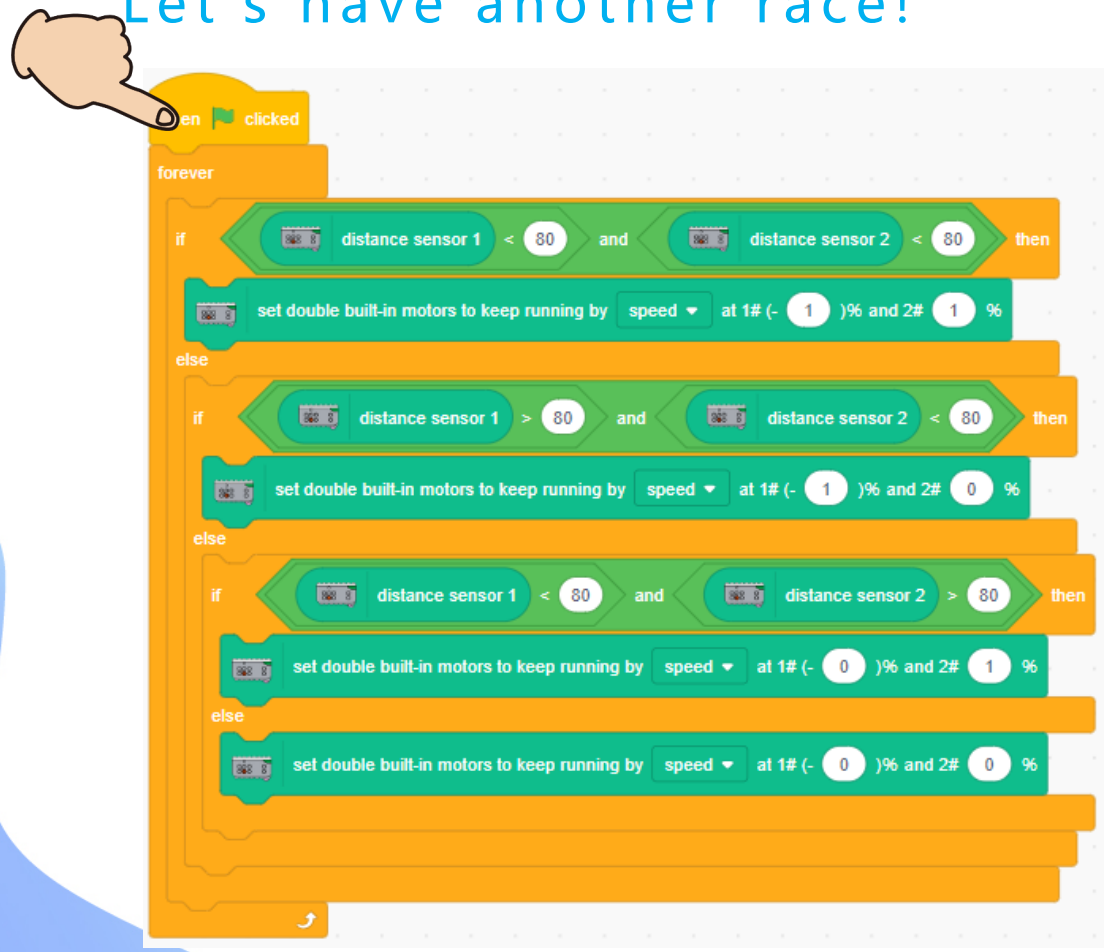
Repeatedly

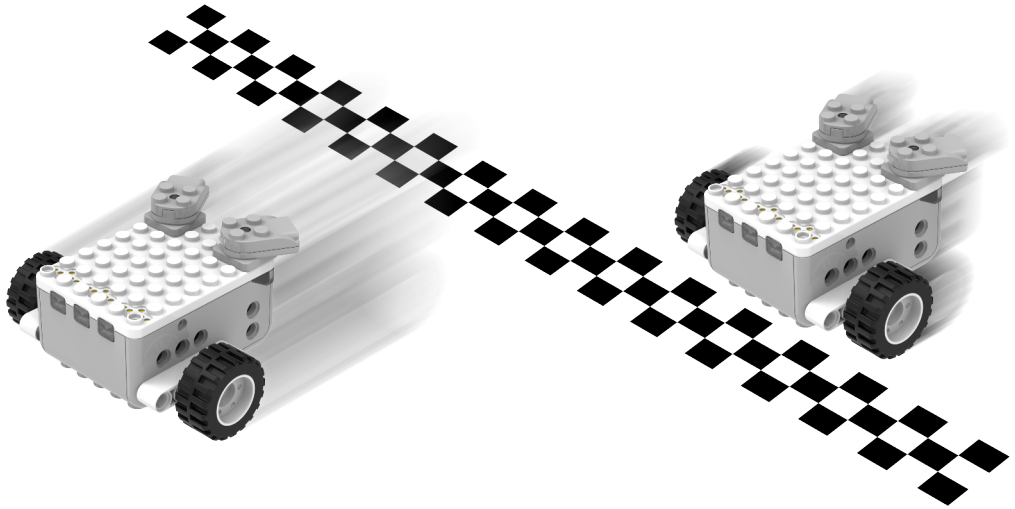


Create

2.Increase the difficulty of the competition.

Let's have another race!

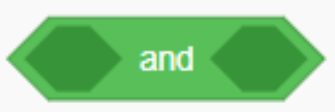




SUMMARY

Summary

1. Logical relationships.



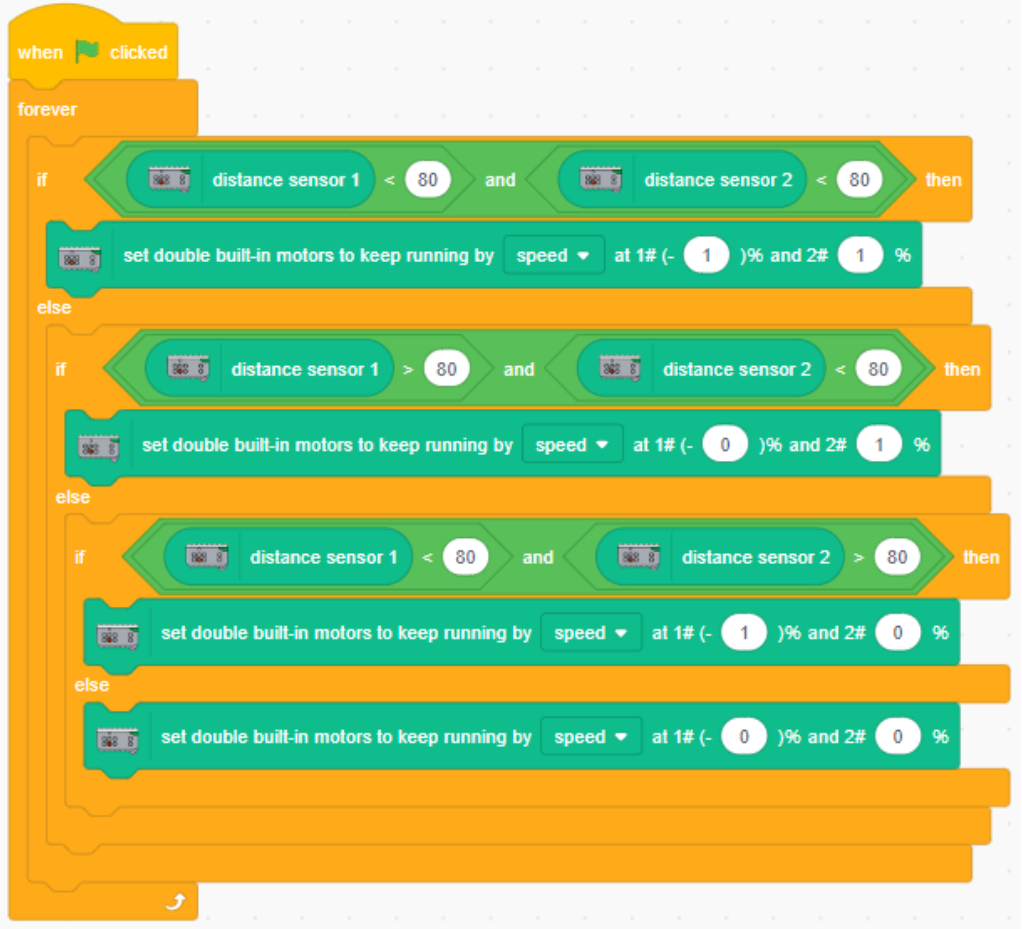
If both conditions are true, output "true"; otherwise, output "false."

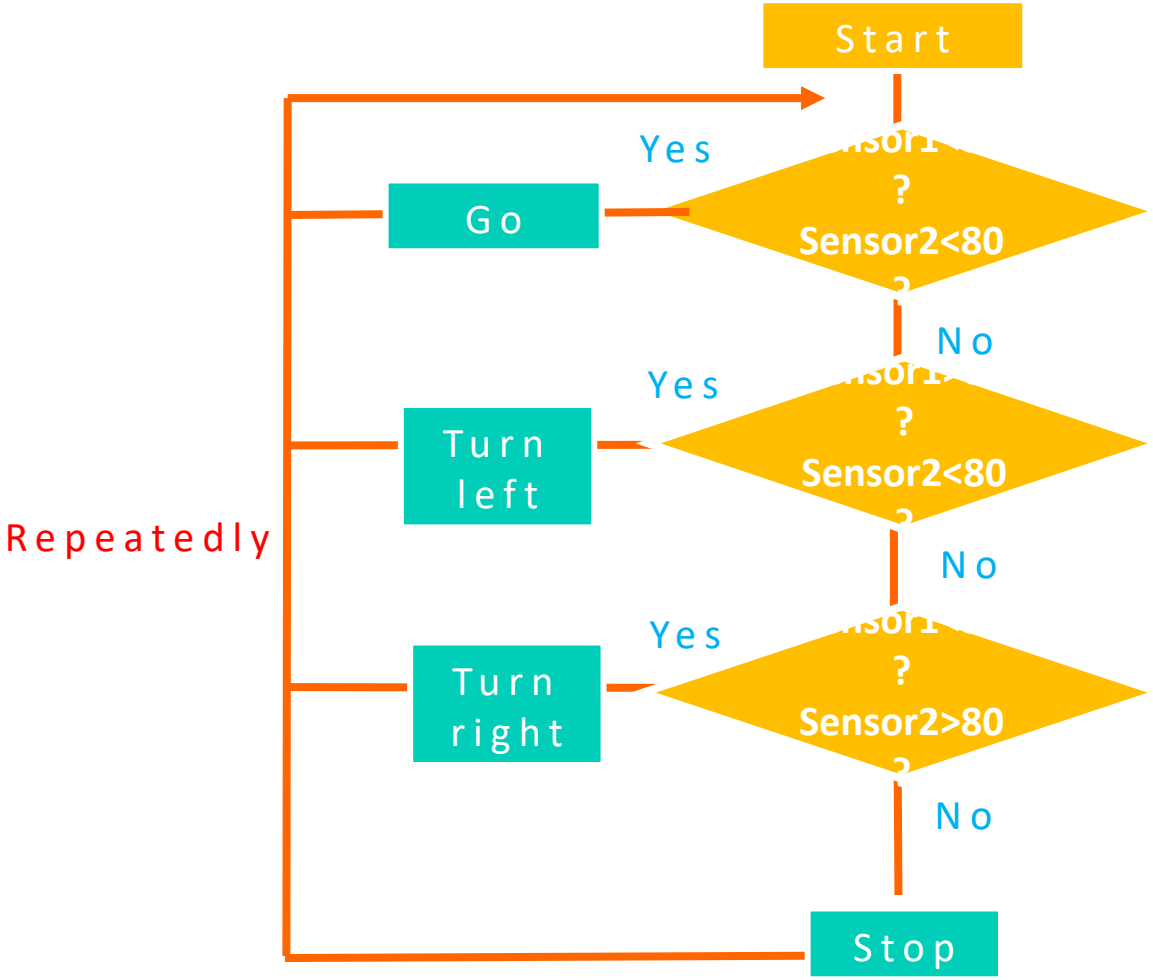
If at least one condition is true, output "true"; if no conditions are true, output "false."

When the selected condition is not met, the output should be "true."

Summary

2. Multibranch structure.

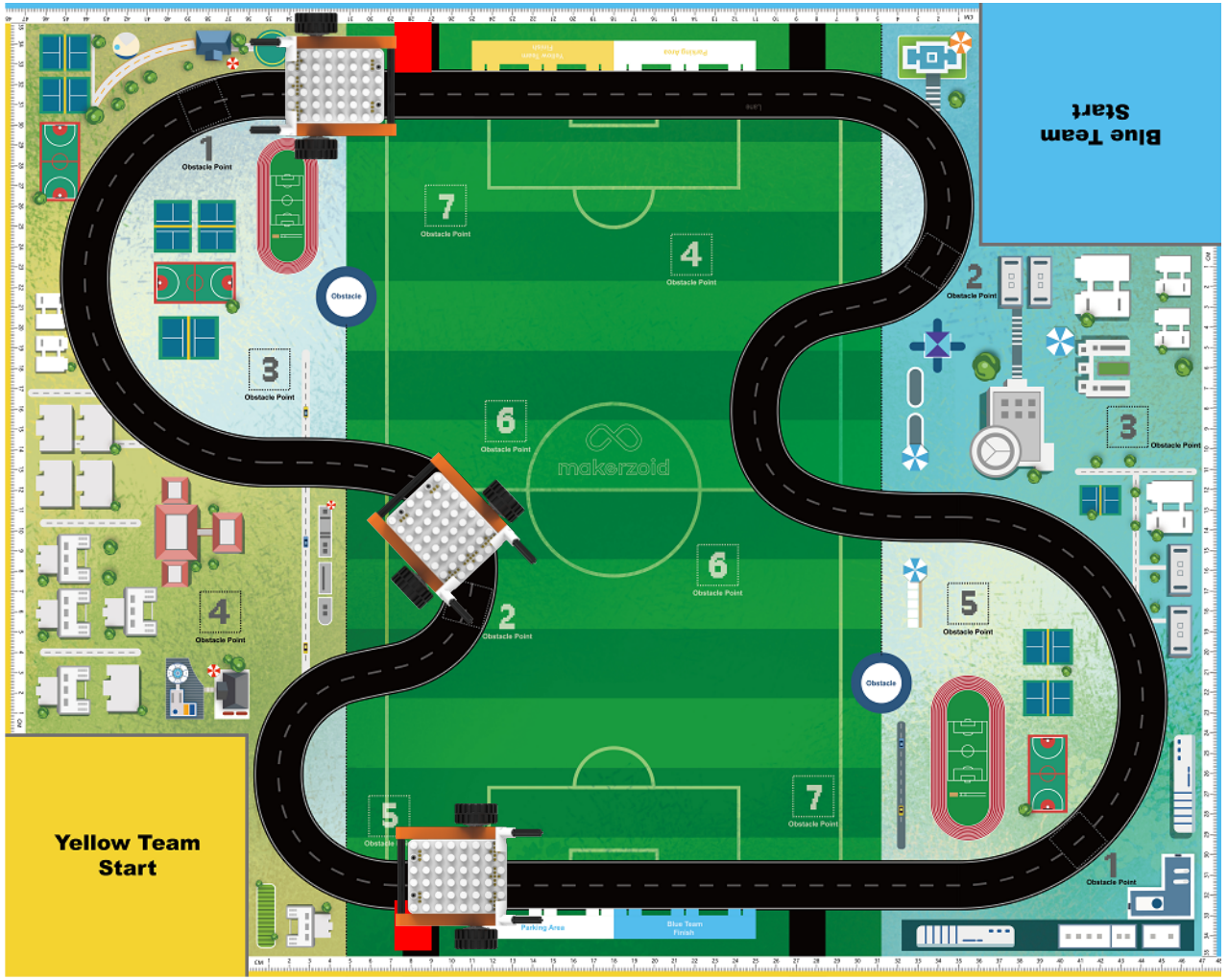




Summary

3. Complete the race.

Finish Point



Start Point

SHARE WITH YOUR PARENTS

Share the knowledge about the Gesture-Controlled Racing with your mom and dad when you get home!