

## For students

1. Using the login info from your teacher, register your student account
2. On your group page, click on your class studio and open the project
3. Click on the Remix button to make a copy of the project template

The screenshot shows the Scratch project page for "SP.248 project" by Timber\_Carey. The "Remix" button is circled in red. The project includes a grid with a robot and a flag, instructions, and notes.

**Instructions**

1. Create a remix of the project
2. Build your algorithm in the robot sprite according to the instructions
3. Submit your project


**Notes and Credits**

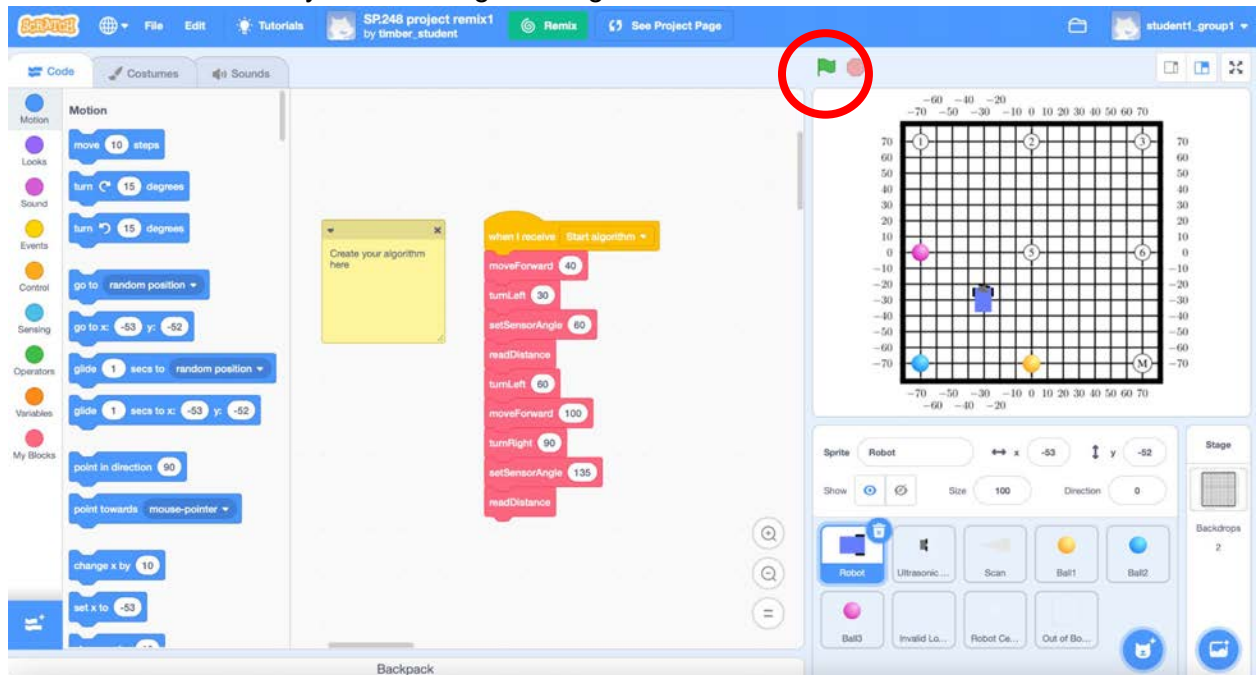
Created by Timber Carey for the class SP.248.

4. In the scripts for the Robot Sprite, find the note that says "Build your algorithm here". You can navigate by dragging the open space with your mouse.

The screenshot shows the Scratch code editor for "SP.248 project remix1" by timber\_student. The code for the robot sprite is visible, and the "Create your algorithm here" note is circled in red.

```
when I receive 'Start algorithm'
  moveForward 40
  turnLeft 30
  setSensorAngle 60
  readDistance
  turnLeft 60
  moveForward 100
  turnRight 90
  setSensorAngle 135
  readDistance
```

- You will create your algorithm under the  block, using blocks from the section on the left. See instructions below for creating your algorithm.
- Do not edit any other blocks of code (i.e. built-in functions on the Robot Sprite Script, code on other sprites, etc.). Do not edit any sprites or backgrounds.
- To test your algorithm while you are making it, you can click the green flag. This will run your algorithm from the start of a round of searching. The ball locations will be randomized each time you click the green flag.



- Once you have made your algorithm, you can submit your project to your instructor.

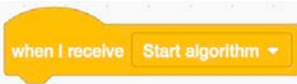
## Creating your algorithm



- Make your algorithm under the block next to the “Build your algorithm here” note. Start by deleting any code blocks that are already there.
- You may only use blocks from the following categories:



- **Control** blocks are used for loops, if statements, and delays. They are not necessary but they help you create more complex and cleaner code.
- **Operators** are used within the Control blocks. They perform operations like math and comparisons. You can nest operators within other operators.
- **Variables** can be used within operator blocks. They can be used to store values like sensor readings from the robot.
  - Do not use or edit the variables that start with “internal:”. These variables are used internally to make the functions work. You can just ignore them.
  - There are some built-in variables that will store values when you use certain robot functions like readDistance. The built-in variables are angle, sensorDistance, xLocation, and yLocation. These variables will only update their values when the corresponding functions are run.
  - You can also create your own variables by clicking the “Make a Variable” button.
- **My Blocks** are built-in functions. These are the main robot functions that you will be using. They do things like make your robot move around the game board and get different sensor readings. Test them out by placing a block under the



block and click the green flag to see what they do.

MIT OpenCourseWare  
<https://ocw.mit.edu/>

SP.248 NEET Ways of Thinking  
Fall 2023

For information about citing these materials or our Terms of Use, visit: <https://ocw.mit.edu/terms>.