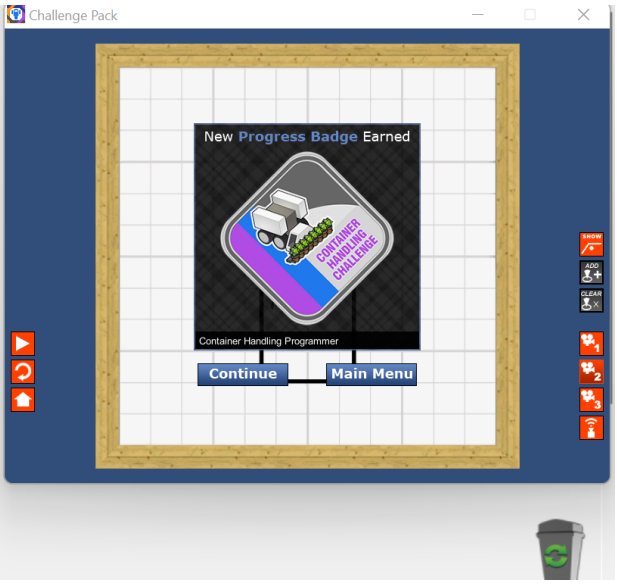
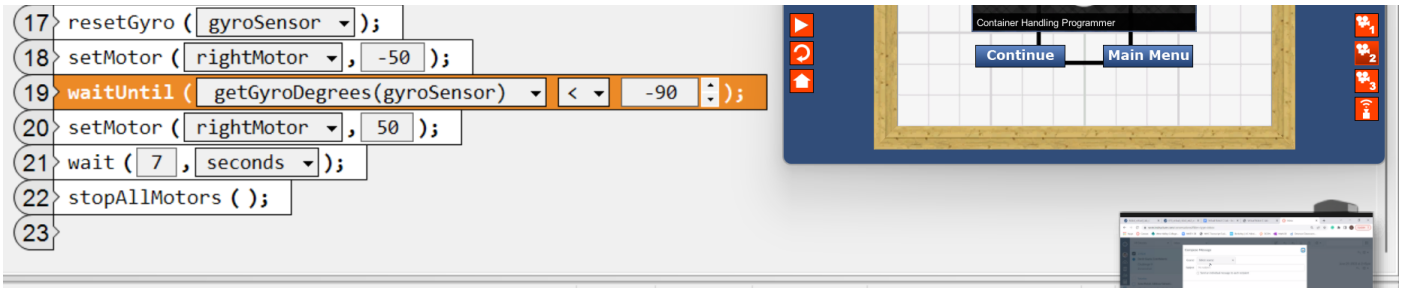
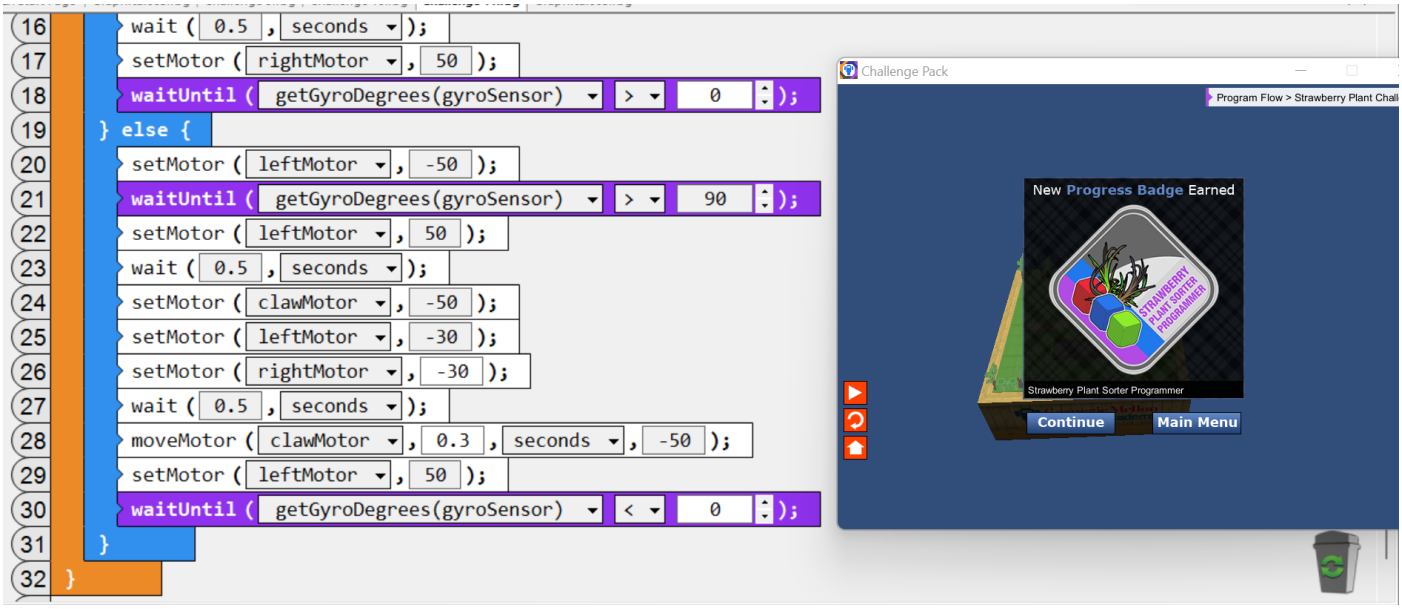
```
1  resetGyro ( gyroSensor );
2  setMotor ( leftMotor , -50 );
3  setMotor ( rightMotor , 50 );
4  waitUntil ( getGyroDegrees(gyroSensor) > 90 );
5  setMotor ( leftMotor , 50 );
6  wait ( 3 , seconds );
7  resetGyro ( gyroSensor );
8  setMotor ( rightMotor , -50 );
9  waitUntil ( getGyroDegrees(gyroSensor) < -90 );
10 setMotor ( rightMotor , 50 );
11 wait ( 8.6 , seconds );
12 resetGyro ( gyroSensor );
13 setMotor ( rightMotor , -50 );
14 waitUntil ( getGyroDegrees(gyroSensor) < -90 );
15 setMotor ( rightMotor , 50 );
16 wait ( 3 , seconds );
```

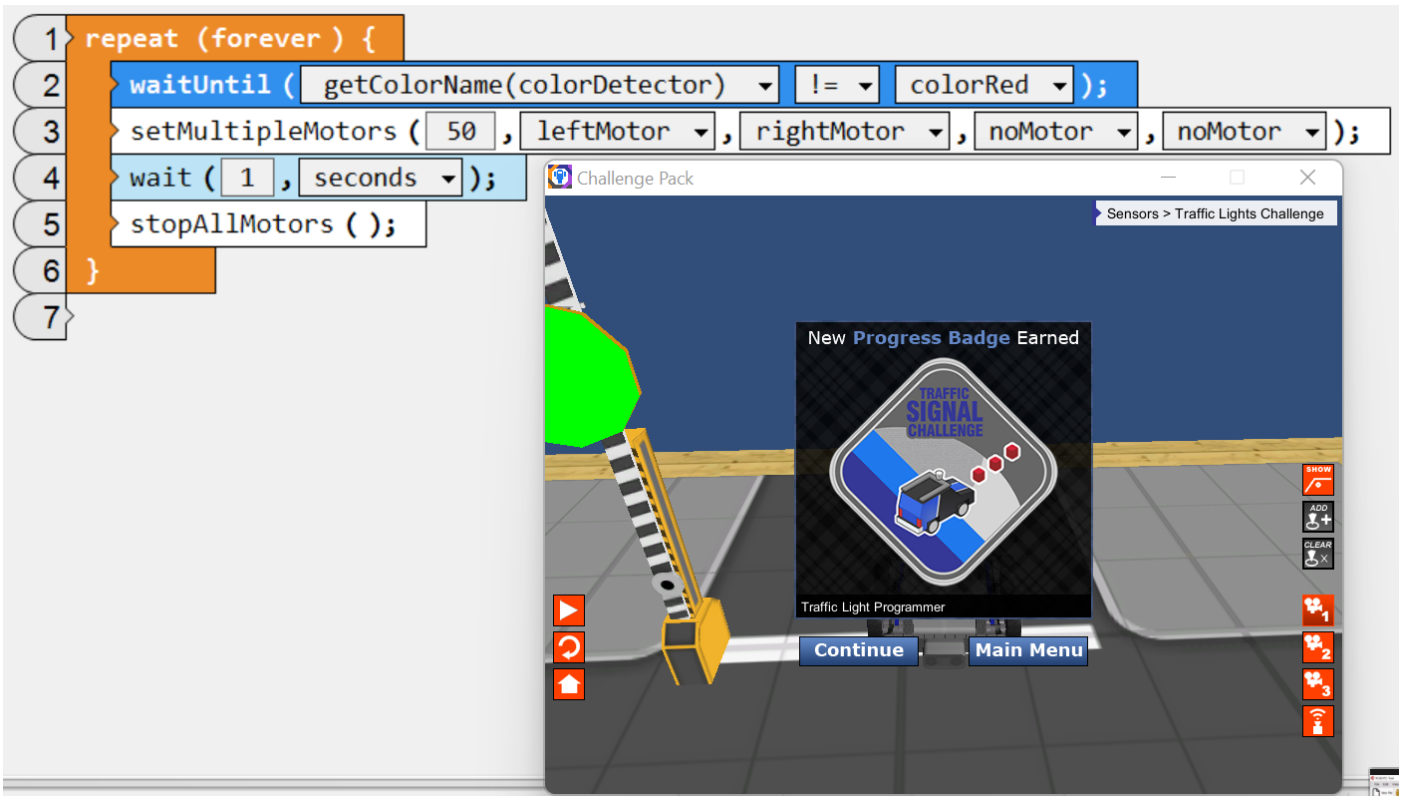


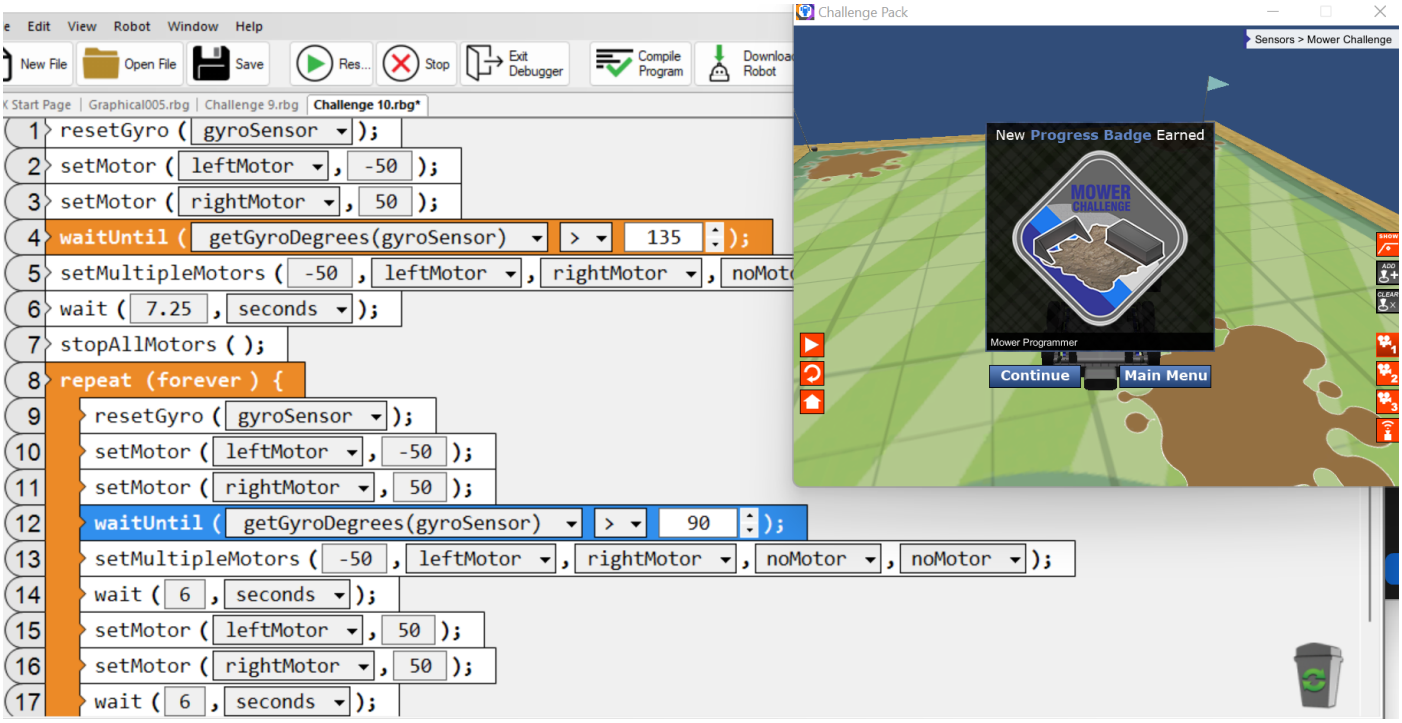

17 resetGyro (gyroSensor);
18 setMotor (rightMotor , -50);
19 waitUntil (getGyroDegrees(gyroSensor) < -90);
20 setMotor (rightMotor , 50);
21 wait (7 , seconds);
22 stopAllMotors ( );
23
Container Handling Programmer
Continue
Main Menu

```
1  setMultipleMotors ( 50 , leftMotor , rightMotor , noMotor , noMotor );
2  wait ( 2.8 , seconds );
3  repeat (forever ) {
4      setMultipleMotors ( 50 , leftMotor , rightMotor , noMotor , noMotor );
5      wait ( 0.8 , seconds );
6      resetGyro ( gyroSensor );
7      moveMotor ( clawMotor , 0.3 , seconds , 50 );
8      if ( getColorName(colorDetector) == colorGreen ) {
9          setMotor ( rightMotor , -50 );
10         waitUntil ( getGyroDegrees(gyroSensor) < -90 );
11         setMotor ( rightMotor , 50 );
12         wait ( 0.5 , seconds );
13         setMotor ( clawMotor , -50 );
14         setMotor ( leftMotor , -50 );
15         setMotor ( rightMotor , -50 );
16         wait ( 0.5 , seconds );
```


wait (0.5 , seconds);
setMotor (rightMotor , 50);
waitUntil (getGyroDegrees(gyroSensor) > 0);
} else {
setMotor (leftMotor , -50);
waitUntil (getGyroDegrees(gyroSensor) > 90);
setMotor (leftMotor , 50);
wait (0.5 , seconds);
setMotor (clawMotor , -50);
setMotor (leftMotor , -30);
setMotor (rightMotor , -30);
wait (0.5 , seconds);
moveMotor (clawMotor , 0.3 , seconds , -50);
setMotor (leftMotor , 50);
waitUntil (getGyroDegrees(gyroSensor) < 0);
}
}
Challenge Pack
Program Flow > Strawberry Plant Chall
New Progress Badge Earned
STRAWBERRY PLANT SORTER PROGRAMMER
Strawberry Plant Sorter Programmer
Continue
Main Menu

```
1 repeat (forever ) {
2     waitUntil ( getColorName(colorDetector) != colorRed );
3     setMultipleMotors ( 50 , leftMotor , rightMotor , noMotor , noMotor );
4     wait ( 1 , seconds );
5     stopAllMotors ( );
6 }
7
```




Edit View Robot Window Help
New File
Open File
Save
Stop
Exit Debugger
Compile Program
Start Page | Graphical005.rbg | Challenge 9.rbg | Challenge 10.rbg*
1 resetGyro (gyroSensor);
2 setMotor (leftMotor , -50);
3 setMotor (rightMotor , 50);
4 waitUntil (getGyroDegrees(gyroSensor) > 135);
5 setMultipleMotors (-50 , leftMotor , rightMotor ,
6 wait (7.25 , seconds);
7 stopAllMotors ( );
8 repeat (forever) {
9 resetGyro (gyroSensor);
10 setMotor (leftMotor , -50);
11 setMotor (rightMotor , 50);
12 waitUntil (getGyroDegrees(gyroSensor) > 90);
13 setMultipleMotors (-50 , leftMotor , rightMotor , noMotor , noMotor);
14 wait (6 , seconds);
15 setMotor (leftMotor , 50);
16 setMotor (rightMotor , 50);
17 wait (6 , seconds);
Challenge Pack
Sensors > Mower Challenge
New Progress Badge Earned
MOWER CHALLENGE
Mower Programmer
Continue
Main Menu