



62880A VR Skills Challenge

Vincent, 62880A
Bayside, NY



COAST 2 COAST
ROBOTICS



**COAST 2 COAST
ROBOTICS**

```

1  #region VEXcode Generated Robot Configuration
2  import math
3  import random
4  from vexcode_vrc import *
5  from vexcode_vrc.events import get_Task_func
6
7  # Brain should be defined by default
8  brain=Brain()
9
10 drivetrain = Drivetrain("drivetrain", 0)
11 arm_motor = Motor("ArmMotor", 3)
12 rotation = Rotation("Rotation", 7)
13 intake_motor = Motor("IntakeMotor", 8)
14 optical = Optical("Optical", 11)
15 gps = GPS("GPS", 20)
16
17 #endregion VEXcode Generated Robot Configuration
18 myVariable = 0
19 message1 = Event()
20 message2 = Event()
21
22 def when_started1():
23     global myVariable, message1, message2
24     # Set all the velocities to max
25     arm_motor.set_velocity(100, PERCENT)
26     intake_motor.set_velocity(100, PERCENT)
27     drivetrain.set_drive_velocity(100, PERCENT)
28     drivetrain.set_turn_velocity(100, PERCENT)
29
30     #Move the robot forward and turn it to shoot the preload over the barrier
31     drivetrain.drive_for(FORWARD, 18, INCHES)
32     drivetrain.turn_for(RIGHT, 90, DEGREES, wait=False)
33     #Shooting preload
34     arm_motor.spin_for(FORWARD, 800, DEGREES)
35     intake_motor.spin(REVERSE)
36     wait(1, SECONDS)
37     #moving arm down to scoop triballs over the barrier
38     arm_motor.spin_for(FORWARD, 400, DEGREES)
39     #scoop the first one
40     drivetrain.drive_for(FORWARD, 30, INCHES)
41     drivetrain.drive_for(REVERSE, 30, INCHES)
42     drivetrain.turn_for(LEFT, 90, DEGREES)
43     drivetrain.drive_for(FORWARD, 17, INCHES)
44
45     #message1 is a function that scoops a triball 30 inches to the right over the barrier
46     message1.broadcast_and_wait()
47     #move to scoop another with message1
48     drivetrain.drive_for(FORWARD, 22, INCHES)
49     message1.broadcast_and_wait()
50     #move to scoop another with message1
51     drivetrain.drive_for(FORWARD, 12, INCHES)
52     message1.broadcast_and_wait()
53     #move to scoop another with message1
54     drivetrain.drive_for(FORWARD, 10, INCHES)
55     message1.broadcast_and_wait()
56     #move the arm out of the way so the robot can run under the alley
57     arm_motor.spin_for(REVERSE, 800, DEGREES, wait=False)
58     drivetrain.drive_for(FORWARD, 36, INCHES)
59     drivetrain.turn_for(RIGHT, 90, DEGREES, wait=False)
60     arm_motor.spin_for(FORWARD, 800, DEGREES)
61     #move through the alley
62     drivetrain.drive_for(FORWARD, 72, INCHES)

```



```

63 intake_motor.spin(FORWARD)
64 drivetrain.drive_for(FORWARD, 12, INCHES)
65 drivetrain.turn_for(LEFT, 45, DEGREES)
66 drivetrain.drive_for(REVERSE, 12, INCHES)
67 #pick up the matchload triball
68 drivetrain.turn_for(RIGHT, 90, DEGREES)
69 drivetrain.drive_for(FORWARD, 20, INCHES)
70 intake_motor.spin(REVERSE)
71 drivetrain.drive_for(FORWARD, 7, INCHES)
72 wait(1, SECONDS)
73 #place the intaked triball into the goal
74 drivetrain.drive_for(REVERSE, 25, INCHES)
75 drivetrain.turn_for(RIGHT, 90, DEGREES)
76 intake_motor.spin(FORWARD)
77 #move in front of the goal to push the scooped triballs into the goal
78 drivetrain.drive_for(FORWARD, 40, INCHES)
79 drivetrain.turn_for(LEFT, 45, DEGREES)
80 drivetrain.drive_for(FORWARD, 7, INCHES)
81 intake_motor.spin(REVERSE)
82 #repeat pushing motion along the entire goal
83 message2.broadcast_and_wait()
84 for repeat_count in range(4):
85     drivetrain.drive_for(FORWARD, 12, INCHES)
86     message2.broadcast_and_wait()
87     wait(5, MSEC)
88
89 #picks up any triball 30 inches to the right of the robot
90 def onevent_message1_0():
91     global myVariable, message1, message2
92     drivetrain.turn_for(RIGHT, 90, DEGREES)
93     drivetrain.drive_for(FORWARD, 30, INCHES)
94     drivetrain.drive_for(REVERSE, 30, INCHES)
95     drivetrain.turn_for(LEFT, 90, DEGREES)
96
97 #pushes all triballs on the left of the robot 30 inches (into the goal)
98 def onevent_message2_0():
99     global myVariable, message1, message2
100     drivetrain.turn_for(LEFT, 90, DEGREES)
101     drivetrain.drive_for(FORWARD, 30, INCHES)
102     wait(1, SECONDS)
103     drivetrain.drive_for(REVERSE, 30, INCHES)
104     drivetrain.turn_for(RIGHT, 90, DEGREES)
105
106 # system event handlers
107 message1(onevent_message1_0)
108 message2(onevent_message2_0)
109 # add 15ms delay to make sure events are registered correctly.
110 wait(15, MSEC)
111
112 vr_thread(when_started1)
113

```

