1.

(a) True

(b) False

(c) False

(d) True

(e) True

(f) False

(g) False

(h) True

(i) True

(j) False

2.

1 #pragma config (Sensor, S3, colour, sensorEV3 Color)

2 #pragma config (Motor, motorB, leftwheel, tmotorEV3 Large, PIDControl, encoder)

3 #pragma config (Motor, motorC, rightwheel, tmotorEV3 Large, PIDControl, decoder) ***encoder\****

4 // ∗!! Code a u t o m a t i c a l l y g e n e r a t e d by ’ROBOTC ’

c o n f i g u r a t i o n w i z a r d!! ∗ /

5

6 int nMotorSpeedSet ting = 30;

7 float nPfactor = 0.3;

8 int grey = 50;

9 int lowest = 100;

10 int highest = 0;

11

12 void scanLine( )

13 {

14 motor [leftWheel] = 10;

15 motor [rightWheel] = −10;

16

17 time1 [t1] = 0; ***no data type for array\****

18 while time1 [T1] < 500) ***no opening bracket\****

19 {

20 if (SensorValue [colour] > highest)

21 {

22 highest = SensorValue [colour];

23}

24 if (SensorValue [colour] < lowest)

25 {

26 lowest = SensorValue [colour];

27}

28\_\_\_\_ ***no closing curly bracket for void scanline()\****

29 grey = (highest − lowest) / 2;

30 motor [leftwheel] = 0;

31 motor [rightwheel] = 0;

32

33 return 1; ***the return type of the function is void\****

34}

35

36 task main ( )

37 {

38 float error;

39 scanLine (int turnLeft); ***improper arguments\****

40 while (t r u e)

41 {

42 SensorValue [colour] − grey; ***not a proper assignment operation***

43 motor (leftwheel] = nMotorSpeedSet ting − round (error ∗nP factor); ***different brackets\****

44 motor [rightwheel] = nMotorSpeedSetting + round (error∗nP factor);

45 wait1Msec (−50); ***random function was never declared\****

46

47}

48}