

PROJETO ROBÓTICA – WEBOTS

Nome : Lucas F. Almeida

RA: 22.120.081-9

Lucas Marques

22.120.066-0

Patrick Mautari

22.218.012-7

Enzo Pellegrini

22.120.077-7

CÓDIGO DO ROBÔ

```
1 #include <stdio.h>
2 #include <webots/robot.h>
3 #include <webots/motor.h>
4 #include <webots/distance_sensor.h>
5 #include <webots/led.h>
6 #include <webots/supervisor.h>
7
8 #define TIME_STEP 256
9 #define QtddSensoresProx 8
10 #define QtddLeds 10
11
12 int main(int argc, char **argv) {
13     int i=0;
14     char texto[256];
15     double LeituraSensorProx[QtddSensoresProx];
16     double AceleradorDireito=1.0, AceleradorEsquerdo=1.0;
17     const double *caixa1PosInicPont, *caixa2PosInicPont, *caixa3PosInicPont, *caixa4PosInicPont, *caixa5PosInicPont, *caixa6Pos:
18     const double *caixa1Pos, *caixa2Pos, *caixa3Pos, *caixa4Pos, *caixa5Pos, *caixa6Pos, *caixa7Pos, *caixa8Pos;
19     double caixa1PosInicX, caixa1PosInicY;
20     double caixa2PosInicX, caixa2PosInicY;
21     double caixa3PosInicX, caixa3PosInicY;
22     double caixa4PosInicX, caixa4PosInicY;
23     double caixa5PosInicX, caixa5PosInicY;
24     double caixa6PosInicX, caixa6PosInicY;
25     double caixa7PosInicX, caixa7PosInicY;
26     double caixa8PosInicX, caixa8PosInicY;
27
28
29     /* necessary to initialize webots stuff */
30     for(i=0;i<256;i++) texto[i]='\0';
31
32     wb_robot_init();
33
34     WbNodeRef caixa1 = wb_supervisor_node_get_from_def("caixa1");
35     WbFieldRef caixa1Trans = wb_supervisor_node_get_field(caixa1, "translation");
36
37     WbNodeRef caixa2 = wb_supervisor_node_get_from_def("caixa2");
38     WbFieldRef caixa2Trans = wb_supervisor_node_get_field(caixa2, "translation");
39
40     WbNodeRef caixa3 = wb_supervisor_node_get_from_def("caixa3");
41     WbFieldRef caixa3Trans = wb_supervisor_node_get_field(caixa3, "translation");
42
43     WbNodeRef caixa4 = wb_supervisor_node_get_from_def("caixa4");
44     WbFieldRef caixa4Trans = wb_supervisor_node_get_field(caixa4, "translation");
45 }
```

```
45
46 WbNodeRef caixa5 = wb_supervisor_node_get_from_def("caixa5");
47 WbFieldRef caixa5Trans = wb_supervisor_node_get_field(caixa5, "translation");
48
49 WbNodeRef caixa6 = wb_supervisor_node_get_from_def("caixa6");
50 WbFieldRef caixa6Trans = wb_supervisor_node_get_field(caixa6, "translation");
51
52 WbNodeRef caixa7 = wb_supervisor_node_get_from_def("caixa7");
53 WbFieldRef caixa7Trans = wb_supervisor_node_get_field(caixa7, "translation");
54
55 WbNodeRef caixa8 = wb_supervisor_node_get_from_def("caixa8");
56 WbFieldRef caixa8Trans = wb_supervisor_node_get_field(caixa8, "translation");
57
58 caixa1PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa1Trans);
59 caixa1PosInicX = caixa1PosInicPont[0];
60 caixa1PosInicY = caixa1PosInicPont[1];
61
62 caixa2PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa2Trans);
63 caixa2PosInicX = caixa2PosInicPont[0];
64 caixa2PosInicY = caixa2PosInicPont[1];
65
66 caixa3PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa3Trans);
67 caixa3PosInicX = caixa3PosInicPont[0];
68 caixa3PosInicY = caixa3PosInicPont[1];
69
70 caixa4PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa4Trans);
71 caixa4PosInicX = caixa4PosInicPont[0];
72 caixa4PosInicY = caixa4PosInicPont[1];
73
74 caixa5PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa5Trans);
75 caixa5PosInicX = caixa5PosInicPont[0];
76 caixa5PosInicY = caixa5PosInicPont[1];
77
78 caixa6PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa6Trans);
79 caixa6PosInicX = caixa6PosInicPont[0];
80 caixa6PosInicY = caixa6PosInicPont[1];
81
82 caixa7PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa7Trans);
83 caixa7PosInicX = caixa7PosInicPont[0];
84 caixa7PosInicY = caixa7PosInicPont[1];
85
86 caixa8PosInicPont = wb_supervisor_field_get_sf_vec3f(caixa8Trans);
```

```

87  caixa8PosInicX = caixa8PosInicPont[0];
88  caixa8PosInicY = caixa8PosInicPont[1];
89
90
91  //configurei MOTORES
92
93  WbDeviceTag MotorEsquerdo, MotorDireito;
94
95  MotorEsquerdo = wb_robot_get_device("left wheel motor");
96  MotorDireito = wb_robot_get_device("right wheel motor");
97
98  wb_motor_set_position(MotorEsquerdo, INFINITY);
99  wb_motor_set_position(MotorDireito, INFINITY);
100
101  wb_motor_set_velocity(MotorEsquerdo,0);
102  wb_motor_set_velocity(MotorDireito,0);
103
104  //configurei Sensores de Proximidade
105
106  WbDeviceTag SensorProx[QtddSensoresProx];
107
108  SensorProx[0] = wb_robot_get_device("ps0");
109  SensorProx[1] = wb_robot_get_device("ps1");
110  SensorProx[2] = wb_robot_get_device("ps2");
111  SensorProx[3] = wb_robot_get_device("ps3");
112  SensorProx[4] = wb_robot_get_device("ps4");
113  SensorProx[5] = wb_robot_get_device("ps5");
114  SensorProx[6] = wb_robot_get_device("ps6");
115  SensorProx[7] = wb_robot_get_device("ps7");
116
117  wb_distance_sensor_enable(SensorProx[0],TIME_STEP);
118  wb_distance_sensor_enable(SensorProx[1],TIME_STEP);
119  wb_distance_sensor_enable(SensorProx[2],TIME_STEP);
120  wb_distance_sensor_enable(SensorProx[3],TIME_STEP);
121  wb_distance_sensor_enable(SensorProx[4],TIME_STEP);
122  wb_distance_sensor_enable(SensorProx[5],TIME_STEP);
123  wb_distance_sensor_enable(SensorProx[6],TIME_STEP);
124  wb_distance_sensor_enable(SensorProx[7],TIME_STEP);
125
126

```

```

126
127 //config Leds
128 WbDeviceTag Leds[QtddLeds];
129 Leds[0] = wb_robot_get_device("led0");
130 Leds[1] = wb_robot_get_device("led1");
131 Leds[2] = wb_robot_get_device("led2");
132 Leds[3] = wb_robot_get_device("led3");
133 Leds[4] = wb_robot_get_device("led4");
134 Leds[5] = wb_robot_get_device("led5");
135 Leds[6] = wb_robot_get_device("led6");
136 Leds[7] = wb_robot_get_device("led7");
137 wb_led_set(Leds[0], -1);
138 wb_led_set(Leds[1], -1);
139 wb_led_set(Leds[2], -1);
140 wb_led_set(Leds[3], -1);
141 wb_led_set(Leds[4], -1);
142 wb_led_set(Leds[5], -1);
143 wb_led_set(Leds[6], -1);
144 wb_led_set(Leds[7], -1);
145
146 while (wb_robot_step(TIME_STEP) != -1) {
147
148
149
150     for(i=0;i<256;i++) texto[i]=0;
151
152     for(i=0;i<QtddSensoresProx;i++){
153         LeituraSensorProx[i]= wb_distance_sensor_get_value(SensorProx[i])-60;
154         //sprintf(texto,"%s%d: %5.2f  ",texto,i,LeituraSensorProx[i]);
155     }
156
157     printf("%s\n",texto);
158
159
160
161     if(LeituraSensorProx[0]>200 || LeituraSensorProx[1]>200 || LeituraSensorProx[2]>200){
162
163
164         AceleradorDireito = 1;
165         AceleradorEsquerdo = -1;
166     }

```

```

167
168     else if(LeituraSensorProx[6]>200 || LeituraSensorProx[7]>200 || LeituraSensorProx[5]>200){
169         AceleradorDireito = -1;
170         AceleradorEsquerdo = 1;
171     }
172
173     else {
174         AceleradorDireito = 1;
175         AceleradorEsquerdo = 1;
176     }
177
178
179     wb_motor_set_velocity(MotorEsquerdo,6.28*AceleradorEsquerdo);
180     wb_motor_set_velocity(MotorDireito, 6.28*AceleradorDireito);
181
182
183     caixa1Pos = wb_supervisor_field_get_sf_vec3f(caixa1Trans);
184     caixa2Pos = wb_supervisor_field_get_sf_vec3f(caixa2Trans);
185     caixa3Pos = wb_supervisor_field_get_sf_vec3f(caixa3Trans);
186     caixa4Pos = wb_supervisor_field_get_sf_vec3f(caixa4Trans);
187     caixa5Pos = wb_supervisor_field_get_sf_vec3f(caixa5Trans);
188     caixa6Pos = wb_supervisor_field_get_sf_vec3f(caixa6Trans);
189     caixa7Pos = wb_supervisor_field_get_sf_vec3f(caixa7Trans);
190     caixa8Pos = wb_supervisor_field_get_sf_vec3f(caixa8Trans);
191
192
193     if(round(caixa1Pos[0]*10000)/10000 != round(caixa1PosInicX*10000)/10000 || round(caixa1Pos[1]*10000)/10000 != round(caixa1PosInicY*10000)/10000){
194         break;
195     }
196
197     if(round(caixa2Pos[0]*10000)/10000 != round(caixa2PosInicX*10000)/10000 || round(caixa2Pos[1]*10000)/10000 != round(caixa2PosInicY*10000)/10000){
198         break;
199     }
200
201     if(round(caixa3Pos[0]*10000)/10000 != round(caixa3PosInicX*10000)/10000 || round(caixa3Pos[1]*10000)/10000 != round(caixa3PosInicY*10000)/10000){
202         break;
203     }
204
205     if(round(caixa4Pos[0]*10000)/10000 != round(caixa4PosInicX*10000)/10000 || round(caixa4Pos[1]*10000)/10000 != round(caixa4PosInicY*10000)/10000){
206         break;
207     }

```

```

208
209     if(round(caixa5Pos[0]*10000)/10000 != round(caixa5PosInicX*10000)/10000 || round(caixa5Pos[1]*10000)/10000 != round(caixa5PosInicY*10000)/10000){
210         break;
211     }
212
213     if(round(caixa6Pos[0]*10000)/10000 != round(caixa6PosInicX*10000)/10000 || round(caixa6Pos[1]*10000)/10000 != round(caixa6PosInicY*10000)/10000){
214         break;
215     }
216
217     if(round(caixa7Pos[0]*10000)/10000 != round(caixa7PosInicX*10000)/10000 || round(caixa7Pos[1]*10000)/10000 != round(caixa7PosInicY*10000)/10000){
218         break;
219     }
220
221     if(round(caixa8Pos[0]*10000)/10000 != round(caixa8PosInicX*10000)/10000 || round(caixa8Pos[1]*10000)/10000 != round(caixa8PosInicY*10000)/10000){
222         break;
223     }
224
225 };
226
227 wb_motor_set_velocity(MotorEsquerdo, 0);
228 wb_motor_set_velocity(MotorDireito, 0);
229
230 wb_led_set(Leds[0], 1);
231 wb_led_set(Leds[1], 1);
232 wb_led_set(Leds[2], 1);
233 wb_led_set(Leds[3], 1);
234 wb_led_set(Leds[4], 1);
235 wb_led_set(Leds[5], 1);
236 wb_led_set(Leds[6], 1);
237 wb_led_set(Leds[7], 1);
238
239 /* Enter your cleanup code here */
240
241 /* This is necessary to cleanup webots resources */
242
243 wb_robot_cleanup();
244
245 return 0;
246
247 }

```

Obs:. Segue um txt junto com a entrega do código copiado, colocamos em forma de print no PDF, pois ao colocarmos com a cópia ele não fica indentado.