```
1: //tasks.c
2:
3: bool drive(float dist, bool direction, bool toStop, int speed,
4:            float &currentdist, int &time, TFileHandle &logfile) {
5:     bool isMoving = true;
6:     string mesg = "";
7:
8:     nMotorEncoder[FDRIVE] = 0;
9:
10:    if(direction) {
11:        motor[FDRIVE] = speed;
12:        motor[RDRIVE] = -speed;
13:    } else {
14:        motor[FDRIVE] = -speed;
15:        motor[RDRIVE] = speed;
16:    }
17:
18:    while((abs(nMotorEncoder[FDRIVE]) <= dist * CONV)) {}
19:
20:    float acc2 = abs(SensorValue[ACCPORT]);
21:    sendLog(logfile, time, mesg, acc2);
22:
23:    if (direction){
24:        if (acc2 > MINACCEL) {
25:            isMoving = true;
26:            currentdist += dist;
27:        }
28:        else {
29:            mesg = "Failed to drive.";
30:            sendLog(logfile, time, mesg);
31:            isMoving = false;
32:        }
33:    }
34:    else {
35:        currentdist -= dist;
36:    }
37:
38:    if(toStop) {
39:        motor[FDRIVE] = motor[RDRIVE] = 0;
40:    }
41:
42:    return isMoving;
43: }
44:
45: void tensionWheels(int &pastRotations, bool spinDown) {
46:    if(!spinDown) {
47:        nMotorEncoder[LDSCREW] = 0;
48:        motor[LDSCREW] = -100;
49:        while(nMotorEncoder[LDSCREW] < LDSCREWROTS){}
50:        motor[LDSCREW] = 0;
51:        pastRotations += nMotorEncoder[LDSCREW];
52:    }
53:    else {
54:        nMotorEncoder[LDSCREW] = 0;
55:        motor[LDSCREW] = 100;
56:        while(abs(nMotorEncoder[LDSCREW]) < pastRotations){}
57:        motor[LDSCREW] = 0;
58:    }
59: }
60:
61: bool clean(float &currentdist, int &time, TFileHandle &logfile) {
```

```
62:     string mesg = "";
63:     int failures = 0;
64:
65:     for(int i = 0; i < HITS; i++) {
66:         mesg = "Started cleaning";
67:         sendLog(logfile, time, mesg);
68:
69:         drive(15, 0, 1, SPEEDLOW, currentdist, time, logfile);
70:         mesg = "Reversing: ";
71:         sendLog(logfile, time, mesg, i);
72:
73:         motor[BRUSH] = 100;
74:         mesg = "Spinning up brush";
75:         sendLog(logfile, time, mesg);
76:
77:         wait1Msec(1000);
78:
79:         while (SensorValue(TOUCHPORT) != 1){
80:             mesg = "Ramming";
81:             sendLog(logfile, time, mesg);
82:             if(!drive(DRIVEDIST, 1, 0, SPEEDRAM, currentdist, time, logfile)){
83:                 failures++;
84:                 mesg = "Clean failures now at: ";
85:                 sendLog(logfile, time, mesg, failures);
86:                 if (failures >= MAXFAIL){
87:                     motor[BRUSH] = 0;
88:                     return false;
89:                 }
90:             }
91:         }
92:         drive(5, 1, 1, SPEEDRAM, currentdist, time, logfile);
93:         wait1Msec(1000);
94:
95:         if(!ultrasonicDist()){
96:             motor[BRUSH] = 0;
97:             return true;
98:     }
99:     }
100:
101:    motor[BRUSH] = 0;
102:    return false;
103: }
104:
105: void escape(float &currentdist, int &time, TFileHandle &logfile) {
106:    bool acceltrue = true;
107:    int failures = 0;
108:
109:    drive(0, 0, 1, 0, currentdist, time, logfile);
110:
111:    while ((currentdist > DISTTOLEAVE) && acceltrue) {
112:        if (!drive(DRIVEDIST, 0, 0, SPEEDHIGH, currentdist, time, logfile)) {
113:            if (failures > MAXFAIL){
114:                acceltrue = false;
115:            }
116:            else {
117:                failures++;
118:            }
119:        }
120:    }
121:
122:    if (!acceltrue) {
```

```
123:         string message = "Mission Failure: Shutting Down.";
124:         sendLog(logfile, time, message);
125:     }
126:     else {
127:         string message = "Escaping.";
128:         sendLog(logfile, time, message);
129:
130:         while(!getButtonPress(buttonAny)) {
131:             drive(DRIVEDIST, 0, 0, SPEEDLOW, currentdist, time, logfile);
132:         }
133:     }
134:     drive(0, 0, 1, 0, currentdist, time, logfile);
135: }
136:
137: void shutdown(int &pastRotations, int &time, TFileHandle &logfile) {
138:     tensionWheels(pastRotations, 1);
139:     string mesg = "Shut down.";
140:     sendLog(logfile, time, mesg);
141: }
142:
```