

```
1: // $Id: keyboard.cpp,v 1.23 2019-02-22 17:40:15-08 - - $
2:
3: // Respond to keystrokes.
4:
5: #include <iostream>
6: #include <string>
7: #include <unordered_map>
8: using namespace std;
9:
10: #include <GL/freeglut.h>
11: #include <libgen.h>
12:
13: unordered_map<int, string> special_keys {
14:     {GLUT_KEY_F1, "GLUT_KEY_F1"},
15:     {GLUT_KEY_F2, "GLUT_KEY_F2"},
16:     {GLUT_KEY_F3, "GLUT_KEY_F3"},
17:     {GLUT_KEY_F4, "GLUT_KEY_F4"},
18:     {GLUT_KEY_F5, "GLUT_KEY_F5"},
19:     {GLUT_KEY_F6, "GLUT_KEY_F6"},
20:     {GLUT_KEY_F7, "GLUT_KEY_F7"},
21:     {GLUT_KEY_F8, "GLUT_KEY_F8"},
22:     {GLUT_KEY_F9, "GLUT_KEY_F9"},
23:     {GLUT_KEY_F10, "GLUT_KEY_F10"},
24:     {GLUT_KEY_F11, "GLUT_KEY_F11"},
25:     {GLUT_KEY_F12, "GLUT_KEY_F12"},
26:     {GLUT_KEY_LEFT, "GLUT_KEY_LEFT"},
27:     {GLUT_KEY_UP, "GLUT_KEY_UP"},
28:     {GLUT_KEY_RIGHT, "GLUT_KEY_RIGHT"},
29:     {GLUT_KEY_DOWN, "GLUT_KEY_DOWN"},
30:     {GLUT_KEY_PAGE_UP, "GLUT_KEY_PAGE_UP"},
31:     {GLUT_KEY_PAGE_DOWN, "GLUT_KEY_PAGE_DOWN"},
32:     {GLUT_KEY_HOME, "GLUT_KEY_HOME"},
33:     {GLUT_KEY_END, "GLUT_KEY_END"},
34:     {GLUT_KEY_INSERT, "GLUT_KEY_INSERT"},
35:     {GLUT_KEY_REPEAT_OFF, "GLUT_KEY_REPEAT_OFF"},
36:     {GLUT_KEY_REPEAT_ON, "GLUT_KEY_REPEAT_ON"},
37:     {GLUT_KEY_REPEAT_DEFAULT, "GLUT_KEY_REPEAT_DEFAULT"},
38:     {GLUT_KEY_NUM_LOCK, "GLUT_KEY_NUM_LOCK"},
39:     {GLUT_KEY_BEGIN, "GLUT_KEY_BEGIN"},
40:     {GLUT_KEY_DELETE, "GLUT_KEY_DELETE"},
41:     {GLUT_KEY_SHIFT_L, "GLUT_KEY_SHIFT_L"},
42:     {GLUT_KEY_SHIFT_R, "GLUT_KEY_SHIFT_R"},
43:     {GLUT_KEY_CTRL_L, "GLUT_KEY_CTRL_L"},
44:     {GLUT_KEY_CTRL_R, "GLUT_KEY_CTRL_R"},
45:     {GLUT_KEY_ALT_L, "GLUT_KEY_ALT_L"},
46:     {GLUT_KEY_ALT_R, "GLUT_KEY_ALT_R"},
47: };
48:
```

```
49:
50: unordered_map<int,string> control_chars {
51:     { 0, "NUL"}, { 1, "SOH"}, { 2, "STX"}, { 3, "ETX"},
52:     { 4, "EOT"}, { 5, "ENQ"}, { 6, "ACK"}, { 7, "BEL"},
53:     { 8, "BS" }, { 9, "HT" }, { 10, "LF" }, { 11, "VT" },
54:     { 12, "FF" }, { 13, "CR" }, { 14, "SO" }, { 15, "SI" },
55:     { 16, "DLE"}, { 17, "DC1"}, { 18, "DC2"}, { 19, "DC3"},
56:     { 20, "DC4"}, { 21, "NAK"}, { 22, "SYN"}, { 23, "ETB"},
57:     { 24, "CAN"}, { 25, "EM" }, { 26, "SUB"}, { 27, "ESC"},
58:     { 28, "FS" }, { 29, "GS" }, { 30, "RS" }, { 31, "US" },
59:     {127, "DEL"},
60: };
61:
62: // Characteristics of the window.
63: struct window {
64:     string name;
65:     int width {256};
66:     int height {192};
67: } window;
68:
69: // Called by glutMainLoop to display window contents.
70: void display() {
71:     cout << __PRETTY_FUNCTION__ << "()" << endl;
72:     glClearColor (0.25, 0.25, 0.25, 1.0);
73:     glClear (GL_COLOR_BUFFER_BIT);
74:     glutSwapBuffers();
75: }
76:
77: void print_special_key (int key) {
78:     cout << key << "(";
79:     const auto& keyname = special_keys.find (key);
80:     if (keyname == special_keys.end()) cout << "Unknown GLUT_KEY";
81:     else cout << keyname->second;
82:     cout << ")";
83: }
84:
85: void print_keyboard_key (int key) {
86:     cout << key << "(";
87:     if (isgraph (key)) cout << "'" << GLubyte (key) << "'";
88:     else {
89:         const auto& control = control_chars.find (key);
90:         if (control != control_chars.end()) cout << control->second;
91:     }
92:     cout << ")";
93: }
94:
```

```
95:
96: void special (int key, int x, int y) {
97:     cout << __PRETTY_FUNCTION__ << ": ";
98:     print_special_key (key);
99:     cout << ", " << x << ", " << y << endl;
100: }
101:
102: void specialup (int key, int x, int y) {
103:     cout << __PRETTY_FUNCTION__ << ": ";
104:     print_special_key (key);
105:     cout << ", " << x << ", " << y << endl;
106: }
107:
108: void keyboard (GLubyte key, int x, int y) {
109:     cout << __PRETTY_FUNCTION__ << ": ";
110:     print_keyboard_key (key);
111:     cout << ", " << x << ", " << y << endl;
112: }
113:
114: void keyboardup (GLubyte key, int x, int y) {
115:     cout << __PRETTY_FUNCTION__ << ": ";
116:     print_keyboard_key (key);
117:     cout << ", " << x << ", " << y << endl;
118: }
119:
```

```
120:
121: void reshape (int width, int height) {
122:     cout << __PRETTY_FUNCTION__ << ": "
123:         << width << ", " << height << endl;
124:     window.width = width;
125:     window.height = height;
126:     glMatrixMode (GL_PROJECTION);
127:     glLoadIdentity();
128:     gluOrtho2D (0, window.width, 0, window.height);
129:     glMatrixMode (GL_MODELVIEW);
130:     glViewport (0, 0, window.width, window.height);
131:     glutPostRedisplay();
132: }
133:
134: int main (int argc, char** argv) {
135:     window.name = basename (argv[0]);
136:     glutInit (&argc, argv);
137:     glutInitDisplayMode (GLUT_RGBA | GLUT_DOUBLE);
138:     glutInitWindowSize (window.width, window.height);
139:     glutCreateWindow (window.name.c_str());
140:     glutDisplayFunc (display);
141:     glutReshapeFunc (reshape);
142:     glutKeyboardFunc (keyboard);
143:     glutKeyboardUpFunc (keyboardup);
144:     glutSpecialFunc (special);
145:     glutSpecialUpFunc (specialup);
146:     glutMainLoop();
147:     return 0;
148: }
149:
150: //TEST// mkpspdf keyboard.ps keyboard.cpp*
151:
```

[illegible]