

GLUT keyboard callbacks

```
void glutKeyboardFunc  (void (*func)(unsigned char key, int x, int y));
void glutKeyboardUpFunc(void (*func)(unsigned char key, int x, int y));

void glutSpecialFunc   (void (*func)(int key, int x, int y));
void glutSpecialUpFunc (void (*func)(int key, int x, int y));
```

Registers with GLUT, for the current window, the named function to be called when a key on the keyboard goes *down* (i.e., is pressed), and *up* (released).

Parameters **x** and **y** contain the location in window coordinates of the mouse cursor when the keyboard event occurred.

Parameter **key** is a value representing the key:

glutKeyboardFunc, glutKeyboardUpFunc: ASCII value
glutSpecialFunc, glutSpecialUpFunc: A constant defined in `glut.h` for the function keys, arrow keys, PageUp/Down, Home, End, and Insert keys.

Function prototypes: `void keydown(unsigned char key, int x, int y);`
`void keyup(unsigned char key, int x, int y);`
`void specialdown(int key, int x, int y);`
`void specialup(int key, int x, int y);`

Register with GLUT: `glutKeyboardFunc(keydown);`
`glutKeyboardUpFunc(keyup);`
`glutSpecialFunc(specialdown);`
`glutSpecialUpFunc(specialup);`

`int glutGetModifiers()` indicates the status of the SHIFT, ALT, or CTRL key by its return value:

GLUT_ACTIVE_SHIFT	- SHIFT key is down or Caps Lock is on.
GLUT_ACTIVE_CTRL	- CTRL key is down.
GLUT_ACTIVE_ALT	- ALT key is down.
0	- none of the above keys is down

The reporting of auto repeated keystrokes can be enabled or disabled using:

```
void glutIgnoreKeyRepeat( int ignore );
```

A non-zero (zero) value for `ignore` indicates auto repeated keystrokes should (should not) be ignored.

If auto repeated keystrokes are not ignored, a GLUT application will experience repeated press/release callbacks (if registered).