for recursion in your hook function does not solve the problem because subsequent hook event calls occur after your hook function has returned.

To break the chain, your hook function needs to modify the graph window only if necessary. For example, assume that your hook function needs to change something when the range of an axis in a graph changes. You need to store the axis range from the last time your function was called using user data (see **GetUserData**). If, in the current call, the axis range has not changed, your function must return without making any changes, thereby breaking the chain.

Keyboard Events

The `WMWinHookStruct` structure has three members used with keyboard and earlyKeyboard events. A fourth member, `focusCtrl`, is used only with earlyKeyboard events, described below.

The `keycode` field works with ASCII characters and some special keys such as keyboard navigation keys.

The `specialKeyCode` fields works with navigation keys, function keys and other special keys. `specialKeyCode` is zero for normal text such as letters, numbers and punctuation.

The `keyText` field works with ASCII characters and non-ASCII characters such as accented characters.

The `specialKeyCode` and `keyText` fields were added in Igor Pro 7. New code that does not need to run with earlier Igor versions should use these new fields instead of the keycode field. See **Keyboard Events Example** on page IV-301 for an example.

Here are the codes for the `specialKeyCode` and `keyCode` fields:

Key	specialKeyCode	keyCode	Note
F1 through F39	1 through 39	Not supported	Function keys
LeftArrow	100	28	
RightArrow	101	29	
UpArrow	102	30	
DownArrow	103	31	
PageUp	104	11	
PageDown	105	12	
Home	106	1	
End	107	4	
Return	200	13	
Enter	201	3	
Tab	202	9	
BackTab	203	Not supported	Tab with Shift pressed
Escape	204	27	
Delete	300	8	Backspace key
ForwardDelete	301	127	
Clear	302	Not supported	
Insert	303	Not supported	
Help	400	Not supported	
Break	401	Not supported	Pause/Break key