Double and Triple-Clicking

Double-clicking a word conventionally selects the entire word. Igor extends this idea a bit. In addition to supporting double-clicking, if you triple-click in a line of text, it selects the entire line. If you drag after triple-clicking, it extends the selection an entire line at a time.

Matching Characters

Igor includes a handy feature to help you check parenthesized expressions. If you double-click a parenthesis, Igor tries to find a matching parenthesis on the same line of text. If it succeeds, it selects all of the text between the parentheses. If it fails, it beeps. Consider the command

```
wave1 = exp(log(x)))
```

If you double-clicked on the first parenthesis, it would select "log(x)". If you double-clicked on the last parenthesis, it would beep because there is no matching parenthesis.

If you double-click in-between adjacent parentheses Igor considers this a click on the outside parenthesis.

Igor does matching if you double-click the following characters:

Left and right parentheses	(xxx)
Left and right brackets	[xxx]
Left and right braces	{xxx}
Plain single quotes	'xxx'
Plain double quotes	"xxx"

Code Comments

The Edit menu for procedures contains two items, Commentize and Decommentize, to help you edit and debug your procedure code when you want to comment out large blocks of code, and later, to remove these comments. Commentize inserts comment symbol at the start of each selected line of text. Decommentize deletes any comment symbols found at beginning of each selected line of text.

Code Marker Comments

You can enter specially-formatted comments in procedure files to mark sections of code and later return to those comments using the Code Markers popup menu in the procedure window's navigation bar.

The navigation bar is visible if Show Navigation Bar is checked in the Editing Behavior tab of the Text Editing section of the Miscellaneous Settings dialog. The Code Markers popup menu appears on the left side of the navigation bar and looks like this this: 📍 .

To get a sense of the purpose of code markers, we will look at the "HDF5 Browser.ipf" procedure file. It is an independent module so you must first execute this:

```
SetIgorOption IndependentModuleDev=1   // Enable editing of independent modules
```

Choose Windows→Procedure Windows→HDF5 Browser.ipf

Click the code markers popup menu and note the list of sections in the file such as "Utility Routines", and "Fill List Routines".

Choose Utility Routines from the popup menu. Igor displays the following line:

```
// *** Utility Routines ***
```