

## inverseErf

### inverseErf (x)

The inverseErf function returns the inverse of the error function.

### Details

The function is calculated using rational approximations in several regions followed by one iteration of Halley's algorithm.

### See Also

The **erf**, **erfc**, **dawson**, and **inverseErfc** functions.

## inverseErfc

### inverseErfc (x)

The inverseErfc function returns the inverse of the complementary error function.

### Details

The function is calculated using rational approximations in several regions followed by one iteration of Halley's algorithm.

### See Also

The **erf**, **erfc**, **erfcw**, **dawson**, and **inverseErf** functions.

## ItemsInList

### ItemsInList(listStr [, listSepStr])

The ItemsInList function returns the number of items in *listStr*. *listStr* should contain items separated by *listSepStr* which typically is ";".

Use ItemsInList to count the number of items in a string containing a list of items separated by a string (usually a single ASCII character), such as those returned by functions like **TraceNameList** or **AnnotationList**, or a line from a delimited text file.

If *listStr* is "" then 0 is returned.

*listSepStr* is optional. If missing, *listSepStr* defaults to ";".

### Details

*listStr* is searched for item strings bound by *listSepStr* on the left and right.

An item can be empty. The lists "abc;def;;ghi" and ";abc;def;;ghi;" have four items (the third item is "").

*listStr* is treated as if it ends with a *listSepStr* even if it doesn't. The search is case-sensitive.

In Igor6, only the first byte of *listSepStr* was used. In Igor7 and later, all bytes are used.

### Examples

```
Print ItemsInList("wave0;wave1;wave1#1;") // prints 3
Print ItemsInList("key1=val1,key2=val2", ",") // prints 2
Print ItemsInList("1 \t 2 \t", "\t") // prints 2
Print ItemsInList(";") // prints 1
Print ItemsInList(";;") // prints 2
Print ItemsInList(";a;") // prints 2
Print ItemsInList(";;;") // prints 3
```

### See Also

The **AddListItem**, **StringFromList**, **FindListItem**, **RemoveListItem**, **RemoveFromList**, **WaveList**, **TraceNameList**, **StringList**, **VariableList**, and **FunctionList** functions.

## j

### j

The j function returns the loop index of the 2nd innermost iterate loop in a macro. Not to be used in a function. iterate loops are archaic and should not be used.