

The output wave from MakeIndex contains indices which can be used to access the elements of the input wave in sorted order. These indices can be used later with IndexSort to sort the input wave and/or to reorder other waves. For example:

```
Make/O data0 = {1, 9, 2, 3}
Make/O/N=4 index
MakeIndex data0, index
Print index           // Prints 0, 2, 3, 1
```

The output values 0, 2, 3, and 1 mean that, to sort data0, you would access element 0, then element 2, then element 3, then element 1. For example, this prints the values of data0 in order:

```
Print data0[0], data0[2], data0[3], data0[1]
```

You can now apply that sequence of access using IndexSort:

```
IndexSort index, data0
Print data0           // Prints 1, 2, 3, 9
```

You can apply the same reordering to another wave using IndexSort:

```
Make/O data1 = {0, 1, 2, 3}
IndexSort index, data1
Print data1           // Prints 0, 2, 3, 1
```

Decimation

If you have a large data set it may be convenient to deal with a smaller but representative number of points. In particular, if you have a graph with millions of points, it probably takes a long time to draw or print the graph. You can do without many of the data points without altering the graph much. Decimation is one way to accomplish this.

There are at least two ways to decimate data:

1. Keep only every nth data value. For example, keep the first value, discard 9, keep the next, discard 9 more, etc. We call this **Decimation by Omission** (see page III-135).
2. Replace every nth data value with the result of some calculation such as averaging or filtering. We call this **Decimation by Smoothing** (see page III-136).

Decimation by Omission

To decimate by omission, create the smaller output wave and use a simple assignment statement (see **Waveform Arithmetic and Assignments** on page II-74) to set their values. For example, If you are decimating by a factor of 10 (omitting 9 out of every 10 values), create an output wave with 1/10th as many points as the input wave.

For example, make a 1000 point test input waveform:

```
Make/O/N=1000 wave0
SetScale x 0, 5, wave0
wave0 = sin(x) + gnoise(.1)
```

Now, make a 100 point waveform to contain the result of the decimation:

```
Make/O/N=100 decimated
SetScale x 0, 5, decimated    // preserve the x range
decimated = wave0[p*10]       // for (p=0;p<100;p+=1) decimated[p] = wave0[p*10]
```

Decimation by omission can be obtained more easily using the Resample operation and dialog by using an interpolation factor of 1 and a decimation factor of (in this case) 10, and a filter length of 1.

```
Duplicate/O wave0, wave0Resampled
Resample/DOWN=10/N=1 wave0Resampled
```