SignalProcessing

RectangleWindow

multiply an array of samples by a rectangular windowing function

Calling Sequence

`RectangleWindow(A)`

Parameters

`A` - *Array* of real or complex numeric values; the signal

Options

- container : *Array*, predefined Array for holding results
- inplace : *truefalse*, specifies that output should overwrite input

Description

- The **`RectangleWindow(A)`** command multiplies the Array **`A`** by the rectangular windowing function and returns the result in an Array having the same length.
- The rectangular windowing function $w(k)$ is defined as follows for a sample with N points.

$$w(k) = 1$$

- It is effectively equivalent to no windowing function.
- Before the code performing the computation runs, **`A`** is converted to datatype **`float[8]`** or **`complex[8]`** if it does not have one of

See Also

SignalProcessing

those datatypes already. For this reason, it is most efficient if **A** has one of these datatypes beforehand. This does not apply if **inplace** is true.

- If the **container**=**C** option is provided, then the results are put into **C** and **C** is returned. With this option, no additional memory is allocated to store the result. The container must be an Array of the same size and datatype as **A**.
- If the **inplace** or **inplace**=true option is provided, then **A** is overwritten with the results. In this case, the **container** option is ignored.

Thread Safety

- The **SignalProcessing[RectangleWindow]** command is thread-safe as of Maple 18.
- For more information on thread safety, see index/threadsafe.

Examples

```
> with(SignalProcessing):
> N := 1024:
> a := GenerateUniform(N, -1, 1)
```

$$a := \begin{bmatrix} 1 .. 1024\ Array \\ Data\ Type: float_8 \\ Storage: rectangular \\ Order: C\_order \end{bmatrix} \quad (1)$$

```
> RectangleWindow(a)
```

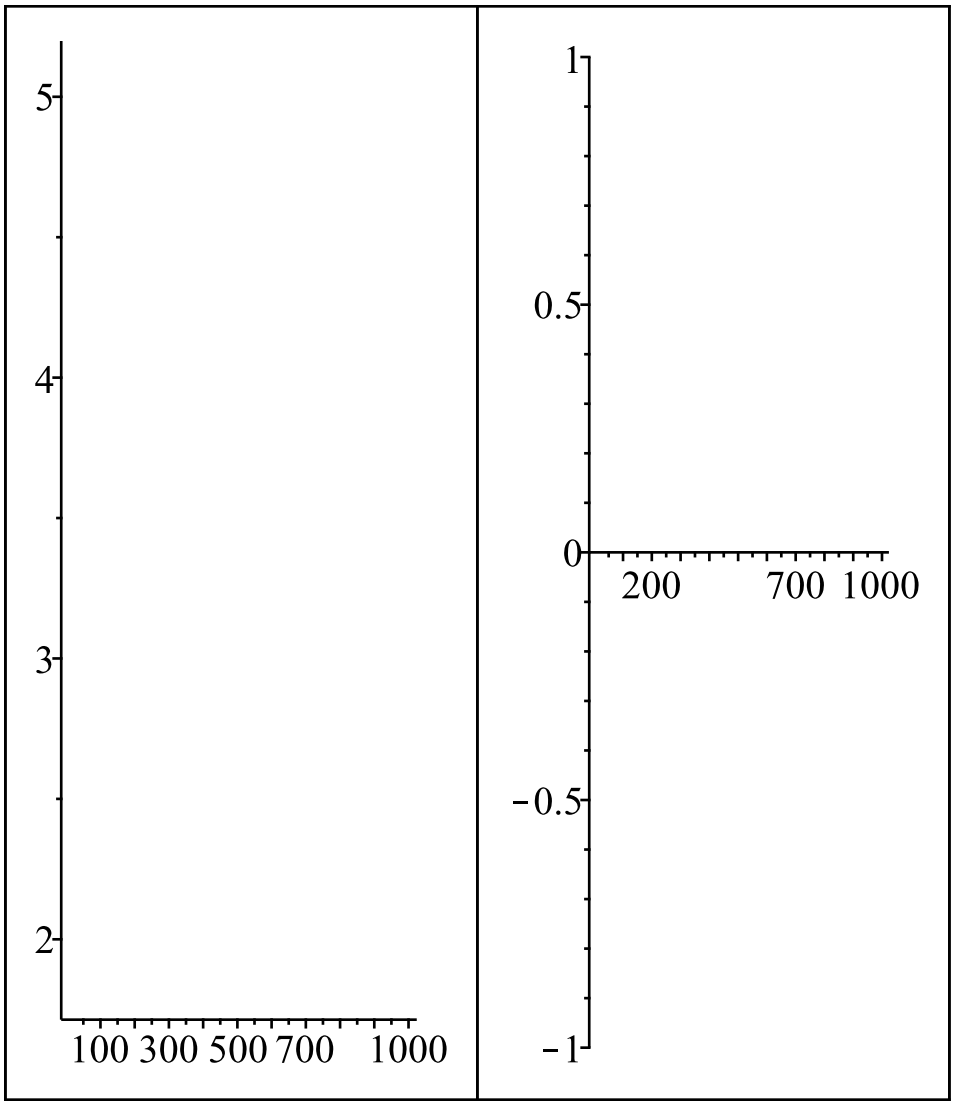
$$\begin{bmatrix} 1 .. 1024\ Vector_{row} \\ Data\ Type: float_8 \\ Storage: rectangular \\ Order: C\_order \end{bmatrix} \quad (2)$$

```
> c := Array(1..N, 'datatype' = 'float'[8], 'order' = 'C_order'):
> RectangleWindow(Array(1..N, 'fill' = 1, 'datatype' = 'float'[8], 'order' =
      'C_order'), 'container' = c)
```

$$\begin{bmatrix} 1 .. 1024\ Vector_{row} \\ Data\ Type: float_8 \\ Storage: rectangular \\ Order: C\_order \end{bmatrix} \quad (3)$$

```
> u := `~`[log](FFT(c)) :
```

```
> use plots in display(Array([listplot(ℜ(u)), listplot(ℑ(u))])) end use
```




```
>
```

Compatibility

- The **SignalProcessing[RectangleWindow]** command was introduced in Maple 18.
- For more information on Maple 18 changes, see Updates in Maple

18.