API Collection

Clipping, limit, and threshold operations

Apply clipping, limit, or threshold rules to the elements in a vector.

Topics

Clipping Operations

The functions in this group restrict the values in a vector so that they fall within a given range or invert values outside a given range.

```
static func clip<U>(U, to: ClosedRange<Double>) -> [Double]
```

Returns the elements of a double-precision vector clipped to the specified range.

```
static func clip<U>(U, to: ClosedRange<Float>) -> [Float]
```

Returns the elements of a single-precision vector clipped to the specified range.

```
static func clip<U, V>(U, to: ClosedRange<Double>, result: inout V)
```

Calculates the elements of a double-precision vector clipped to the specified range.

```
static func clip<U, V>(U, to: ClosedRange<Float>, result: inout V)
```

Calculates the elements of a single-precision vector clipped to the specified range.

```
static func invertedClip<U>(U, to: ClosedRange<Double>) -> [Double]
```

Returns a double-precision vector that's inverted-clipped to the specified range.

```
static func invertedClip<U>(U, to: ClosedRange<Float>) -> [Float]
```

Returns a single-precision vector that's inverted-clipped to the specified range.

```
static func invertedClip<U, V>(U, to: ClosedRange<Double>, result:
inout V)
```

Calculates a double-precision vector that's inverted-clipped to the specified range.

```
static func invertedClip<U, V>(U, to: ClosedRange<Float>, result: inout V)
```

Calculates a single-precision vector that's inverted-clipped to the specified range.

`vDSP_vclip`

Calculates the elements of a single-precision vector clipped to the specified range.

`vDSP_vclipD`

Calculates the elements of a double-precision vector clipped to the specified range.

`vDSP_vclipc`

Calculates and counts the elements of a single-precision vector clipped to the specified range.

`vDSP_vclipcD`

Calculates and counts the elements of a double-precision vector clipped to the specified range.

`vDSP_viclip`

Calculates the elements of a single-precision vector inverted-clipped to the specified range using the specified stride.

`vDSP_viclipD`

Calculates the elements of a double-precision vector inverted-clipped to the specified range using the specified stride.

`vDSP_vthr`

Calculates single-precision vector threshold to the specified range.

`vDSP_vthrD`

Calculates double-precision vector threshold to the specified range.

Limit Operations

```
static func limit<U>(U, limit: Double, withOutputConstant: Double) -> [Double]
```

Returns the double-precision vector test limit.

```
static func limit<U>(U, limit: Float, withOutputConstant: Float) -> [Float]
```

Returns the single-precision vector test limit.

`static func limit<U, V>(U, limit: Double, withOutputConstant: Double, result: inout V)`

Calculates the double-precision vector test limit.

`static func limit<U, V>(U, limit: Float, withOutputConstant: Float, result: inout V)`

Calculates the single-precision vector test limit.

`vDSP_vlim`

Calculates the single-precision vector test limit using the specified stride.

`vDSP_vlimD`

Calculates the double-precision vector test limit using the specified stride.

Threshold Operations

`static func threshold<U>(U, to: Double, with: vDSP.ThresholdRule<Double>) -> [Double]`

Returns the elements of the supplied double-precision vector after applying a specified thresholding rule.

`static func threshold<U>(U, to: Float, with: vDSP.ThresholdRule<Float>) -> [Float]`

Returns the elements of the supplied single-precision vector after applying a specified thresholding rule.

`static func threshold<U, V>(U, to: Double, with: vDSP.ThresholdRule<Double>, result: inout V)`

Calculates the elements of the supplied double-precision vector after applying a specified thresholding rule.

`static func threshold<U, V>(U, to: Float, with: vDSP.ThresholdRule<Float>, result: inout V)`

Calculates the elements of the supplied single-precision vector after applying a specified thresholding rule.

`enum ThresholdRule`

Constants that specify vector threshold rules.

`vDSP_vthres`

Calculates single-precision vector threshold with zero fill to the specified range.

`vDSP_vthresD`

Calculates double-precision vector threshold with zero fill to the specified range.

`vDSP_vthrsc`

Calculates single-precision vector threshold with signed constant to the specified range.

`vDSP_vthrscD`

Calculates double-precision vector threshold with signed constant to the specified range.

See Also

Single-vector arithmetic functions

- Absolute and negation functions

  Compute the absolute or negated value of each element in a vector.

- Integration functions

  Compute the running sum, Simpson, or trapezoidal integration of a vector.

- Normalization functions

  Compute the mean and standard deviation of a vector and calculate new elements to have a zero mean and a unit standard deviation.

- Phase computation functions

  Calculate the element-wise phase values, in radians, of a complex vector.

- Complex conjugation functions

  Calculate the complex conjugate of the elements in a vector.

- Vector squaring functions

  Compute the square, signed square, or squared magnitude of the elements in a vector.

- Fractional part extraction

  Truncate the elements of a vector to a fraction.

- Zero crossing search

Count and find the zero crossings in a vector.