

[Accelerate](#) / [vDSP](#) / `divide(_:_:)`

Type Method

divide(_:_:)

Calculates the single-precision element-wise division of a vector and a scalar value.

iOS 13.0+ | iPadOS 13.0+ | Mac Catalyst | macOS 10.15+ | tvOS 13.0+ | visionOS | watchOS 6.0+

```
static func divide<U>(
    _ vector: U,
    _ scalar: Float
) -> [Float] where U : AccelerateBuffer, U.Element == Float
```

Parameters

vector

The input vector, A.

scalar

The input scalar value, B.

Return Value

The output vector, C.

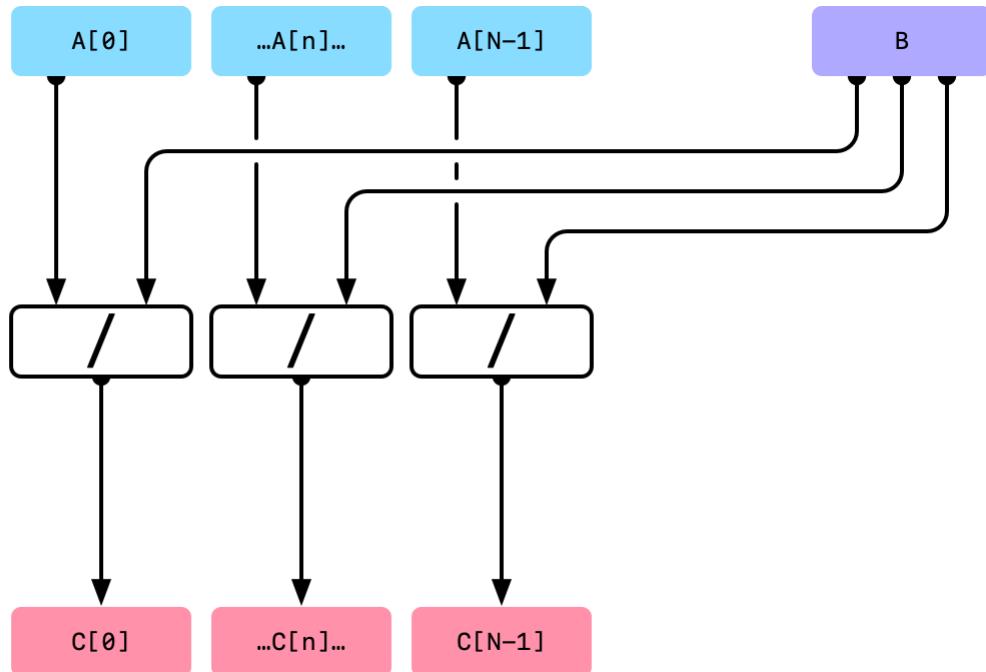
Mentioned in

 [Using vDSP for vector-based arithmetic](#)

Discussion

This function calculates the element-wise division of vector A and scalar value B, and writes the result to vector C.

```
for (n = 0; n < N; ++n)
    C[n] = A[n] / B;
```



The following code shows an example of using this function:

```
let a: [Float] = [1, 2, 3, 4, 5]
let b: Float = 10

let c = vDSP.divide(a, b)

// Prints "[0.1, 0.2, 0.3, 0.4, 0.5]".
print(c)
```

See Also

Division

```
static func divide<U>(Double, U) -> [Double]
```

Returns the double-precision element-wise division of a scalar value and a vector.

```
static func divide<U>(U, Double) -> [Double]
```

Calculates the double-precision element-wise division of a vector and a scalar value.

```
static func divide<T, U>(T, U) -> [Double]
```

Returns the double-precision element-wise division of two vectors.

```
static func divide<U>(Float, U) -> [Float]
```

Returns the single-precision element-wise division of a scalar value and a vector.

```
static func divide<T, U>(T, U) -> [Float]
```

Returns the single-precision element-wise division of two vectors.

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

Calculates the double-precision element-wise division of a scalar value and a vector.

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

Calculates the single-precision element-wise division of a scalar value and a vector.

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

Calculates the double-precision element-wise division of a vector and a scalar value.

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

Calculates the single-precision element-wise division of a vector and a scalar value.

```
static func divide<T, U, V>(T, U, result: inout V)
```

Calculates the double-precision element-wise division of two vectors.

```
static func divide<T, U, V>(T, U, result: inout V)
```

Calculates the single-precision element-wise division of two vectors.

```
static func divide(DSPSplitComplex, by: DSPSplitComplex, count: Int,  
result: inout DSPSplitComplex)
```

Calculates the single-precision elementwise division of a complex vector by a complex vector.

```
static func divide(DSPDoubleSplitComplex, by: DSPDoubleSplitComplex,  
count: Int, result: inout DSPDoubleSplitComplex)
```

Calculates the double-precision elementwise division of a complex vector by a complex vector.

```
static func divide<U>(DSPSplitComplex, by: U, result: inout DSPSplit  
Complex)
```

Calculates the single-precision elementwise division of a complex vector by a real vector.

```
static func divide<U>(DSPDoubleSplitComplex, by: U, result: inout  
DSPDoubleSplitComplex)
```

Calculates the double-precision elementwise division of a complex vector by a complex vector.