Title

sublowertriangle( ) — Return a matrix with zeros above a diagonal

Syntax

```
numeric matrix  sublowertriangle(numeric matrix A [, numeric scalar p])

void           _sublowertriangle(numeric matrix A [, numeric scalar p])
```

where argument *p* is optional.

Description

`sublowertriangle(`*A*`,` *p*`)` returns *A* with the elements above a diagonal set to zero. In the returned matrix, $A[i, j] = 0$ for all $i - j < p$. If it is not specified, *p* is set to zero.

`_sublowertriangle()` mirrors `sublowertriangle()` but modifies *A*. `_sublowertriangle(`*A*`,` *p*`)` sets $A[i, j] = 0$ for all $i - j < p$. If it is not specified, *p* is set to zero.

Remarks and examples

Remarks are presented under the following headings:

Get lower triangle of a matrix

If *A* is a square matrix, then `sublowertriangle(`*A*`, 0)` = `lowertriangle(`*A*`)`. `sublowertriangle()` is a generalization of `lowertriangle()`.

We begin by defining `A`

```
: A = (1, 2, 3 \ 4, 5, 6 \ 7, 8, 9)
```

`sublowertriangle(A, 0)` returns `A` with zeros above the main diagonal as does `lowertriangle()`:

```
: sublowertriangle(A, 0)
        1   2   3
    +-------------+
  1 |   1   0   0 |
  2 |   4   5   0 |
  3 |   7   8   9 |
    +-------------+
```

sublowertriangle(A, 1) returns A with zeros in the main diagonal and above.

```
: sublowertriangle(A, 1)
1   2   3
    +-------------+
  1 |  0   0   0  |
  2 |  4   0   0  |
  3 |  7   8   0  |
    +-------------+
```

sublowertriangle(A, p) can take negative *p*. For example, setting p $= -1$ yields

```
: sublowertriangle(A, -1)
       1   2   3
    +-------------+
  1 |  1   2   0  |
  2 |  4   5   6  |
  3 |  7   8   9  |
    +-------------+
```

Nonsquare matrices

sublowertriangle() and _sublowertriangle() may be used with nonsquare matrices.

For instance, we define a nonsquare matrix A

```
: A = (1, 2, 3, 4 \ 5, 6, 7,  8 \ 9, 10, 11, 12)
```

We use sublowertriangle() to obtain the lower triangle of A:

```
: sublowertriangle(A, 0)
        1    2    3    4
    +---------------------+
  1 |  1    0    0    0   |
  2 |  5    6    0    0   |
  3 |  9   10   11    0   |
    +---------------------+
```

Conformability

sublowertriangle(*A*, *p*):

input:

A: $r \times c$

p: 1×1 (optional)

output:

result: $r \times c$

_sublowertriangle(*A*, *p*):

input:

A: $r \times c$

p: 1×1 (optional)

output:

A: $r \times c$

Diagnostics

None.

Also see

[M-4] **manipulation** — Matrix manipulation