Title

substr( ) — Extract substring

Syntax   Description   Conformability   Diagnostics   Also see

Syntax

string matrix   substr(*string matrix s*, *real matrix b*, *real matrix l*)

string matrix   substr(*string matrix s*, *real matrix b*)

Description

substr(*s*, *b*, *l*) returns the substring of *s* starting at position *b* and continuing for a length of *l*, where

1. *b* specifies the starting position; the first character of the string is $b = 1$.
2. $b > 0$ is interpreted as distance from the start of the string; $b = 2$ means starting at the second character.
3. $b < 0$ is interpreted as distance from the end of string; $b = -1$ means starting at the last character; $b = -2$ means starting at the second from the last character.
4. *l* specifies the length; $l = 2$ means for two characters.
5. $l < 0$ is treated the same as $l = 0$: no characters are copied.
6. $l \geq$. is interpreted to mean to the end of the string.

substr(*s*, *b*) is equivalent to substr(*s*, *b*, .) for strings that do not contain binary 0. If there is a binary 0 to the right of *b*, the substring from *b* up to but not including the binary 0 is returned.

When arguments are not scalar, substr() returns element-by-element results.

Conformability

substr(*s*, *b*, *l*):

s:	$r_1 \times c_1$
b:	$r_2 \times c_2$
l:	$r_3 \times c_3$; *s*, *b*, and *l* r-conformable
result:	$\max(r_1, r_2, r_3) \times \max(c_1, c_2, c_3)$

substr(*s*, *b*):

s:	$r_1 \times c_1$
b:	$r_2 \times c_2$; *s* and *b* r-conformable
result:	$\max(r_1, r_2) \times \max(c_1, c_2)$

Diagnostics

In `substr(`s`, `b`, `l`)` and `substr(`s`, `b`)`, if b describes a position before the beginning of the string or after the end, `""` is returned. If $b + l$ describes a position to the right of the end of the string, results are as if a smaller value for l were specified.

Also see

[M-4] **string** — String manipulation functions