

字符串数组的创建、基本操作和运算方法

讲义和代码可以看配套的课程的第一个视频下载

配套课程b站：《MATLAB教程新手入门篇（数学建模清风主讲，适合零基础同学观看）》
数学建模清风老师版权所有

字符串数组的创建方法

从MATLAB2017a版本开始，我们可以使用英文的双引号来创建字符串

(1) 使用 strings 函数创建空的字符串数组

(2) 使用 string 函数将其他数据类型转换为字符串数组

拓展一：将文本转换为数值数组

拓展二：混合创建字符串数组

拓展三：计算字符串包含的字符数量

拓展四：不推荐的字符串元胞数组

s4 = ["good","great","perfect"] % s4 是包含三个元素的字符串向量	1×3 string 数组 "good" "great" "perfect"
size(s4)	1 3
s5 = ["yes","no"; "op","if"; "and","of"] % s5 是五行两列的字符串矩阵	3×2 string 数组 "yes" "no" "op" "if" "and" "of"
[r,c] = size(s5)	r = 3 c = 2
length(s5) % 行和列较大值	3
numel(s5) % 数组中包含的元素个数	6

ss1 = strings(2,3) % 也可以写成 strings([2,3])	2×3 string 数组 "" "" ""
ss2 = ["", "", " ", " ", " "] % ss2 是我们手动创建的空的字符串数组	2×3 string 数组 "" "" "" "" "" "" "" "" ""
strcmp(ss1,ss2) % 比较数组中对应位置的元素是否相同	2×3 logical 数组 1 1 1 1 1 1
isequal(ss1,ss2) % 比较两个数组整体是否等效	logical 1

c1 = '你好呀'; % 字符串向量 string(c1)	"你好呀"
c2 = char('good','great'); % 字符矩阵 s2 = string(c2)	2×1 string 数组 "good" "great"
strip(s2) % 删除空白字符 % 也能使用 strtrim 或 debblank 函数	2×1 string 数组 "good" "great"
c3 = {'good','great'}; % 字符串元胞数组 s3 = string(c3)	1×2 string 数组 "good" "great"

(1) 使用 str2num函数转换字符串数组或者字符串标量
(2) 使用 double函数转换字符串数组
(3) 使用str2double函数进行转换：str2double函数支持的数据类型比前两个函数更多，它支持字符串数组、字符串元胞数组和字符串数组三种数据类型

在创建字符串数组时，MATLAB允许我们将数值、字符串、逻辑值以及字符串元胞数组混合在一起创建。注意：创建时至少有一个元素需要是字符串类型，下面来看例子：

s1 = ["abc","ab",100]	1×3 string 数组 "abc" "ab" "100"
s2 = [1,2,3; "yes",true,"no"]	2×3 string 数组 "1" "2" "3" "yes" "true" "no"
s3 = [{'你好','我好'},"大家好"]	1×3 string 数组 "你好" "我好" "大家好"

MATLAB在2016b版本中引入了一个专门的函数：strlength。

s1 = "Hello,你好!"; strlength(s1)	9
s2 = ["yes","no", "sorry"; "or", "if", "good"]; strlength(s2) % 计算每个字符串元素的字符数	3 2 5 2 2 4
s3 = strings(3); % 3 行 3 列的空字符串数组 strlength(s3)	0 0 0 0 0 0 0 0 0

当元胞数组中的每个数据都是字符串标量时，这样的元胞数组被称为字符串元胞数组

在MATLAB中，不推荐使用这种类型来保存文本数据。许多文本处理相关的函数不支持字符串元胞数组类型

使用小括号()引用返回的是字符串类型

s1 = ["1 一","5 五","9 九"; "2 二","6 六","10 十"; "3 三","7 七","11 十一"; "4 四","8 八","12 十二"];	4×3 string 数组 "1 一" "5 五" "9 九" "2 二" "6 六" "10 十" "3 三" "7 七" "11 十一" "4 四" "8 八" "12 十二"
s1(3,2)	"7 七"
s1(7) % 线性索引	"7 七"
s1(2:2:end,:) % 偶数行	2×3 string 数组 "2 二" "6 六" "10 十" "4 四" "8 八" "12 十二"
s1([1,4],[2,3]) % 一四行、二三列	2×2 string 数组 "5 五" "9 九" "8 八" "12 十二"

使用大括号{}引用返回的是字符串向量类型

s1{3,2} % 等价于 char(s1(3,2))	'7 七'
[a,b,c] = s1{1,:}	a = '1 一' b = '5 五' c = '9 九'
% 将返回的字符串向量进行水平拼接 [s1{1,:}]	'1 一5 五9 九'
% 将返回的字符串向量保存列到字符串矩阵 char(s1{1,:})	3×2 char 数组 '1 一' '5 五' '9 九'

reverse 函数在 2016b 版本中推出，它能反转文本中的字符顺序。它支持字符串、字符串元胞数组以及字符串数组类型：

reverse(s2)	"gfedcba"
reverse(s3)	1×3 string 数组 "好你" "好我" "好家大"

使用小括号()修改字符串数组时，等号右侧的值可以是字符串、字符串元胞数组或字符串元胞数组，只要大小兼容即可。

s1 = ["abc", "ABC"; "清风", "老师"]; s1(2,2) = "弟弟";	2×2 string 数组 "abc" "ABC" "清风" "弟弟"
s1(2,2) = '哥哥'	2×2 string 数组 "abc" "ABC" "清风" "哥哥"
s1(2,:) = {'666';'888'} % 大小兼容即可	2×2 string 数组 "abc" "ABC" "666" "888"
s1(1,:) = '可爱' % 大小兼容即可	2×2 string 数组 "可爱" "可爱" "666" "888"

然而，当使用大括号{}对字符串数组的元素进行修改时，等号右侧的值只能是字符串向量。

s1 = ["abc", "ABC"; "清风", "老师"]; s1{2,2} = "弟弟"	报错：对字符串进行花括号赋值需要字符串向量。
s1{2,2} = '弟弟'	2×2 string 数组 "abc" "ABC" "清风" "弟弟"

要删除字符串数组中的元素，可以将等号右侧改为空向量[]。

使用小括号()进行删除操作时会改变数组中元素的数量；使用大括号{}则不会改变数组中元素的数量，被删除的位置元素会被缺失字符串代替。

s2 = ["ab","cd","ef"; "gh","ij","kl"; "mn","op","qr"; "st","uv","wx"];	2×3 string 数组 "ab" "cd" "ef" "gh" "ij" "kl" "mn" "op" "qr"
s2([1,4],:) = [] % 先删除 s2 的一四行	
s2(4) = [] % 再删除线性索引为 4 的位置	1×5 string 数组 "gh" "mn" "ij" "kl" "qr"
s2{2} = [] % 使用大括号删除	1×5 string 数组 "gh" <missing> "ij" "kl" "qr"

在 MATLAB 中，我们可以使用加号(+)对字符串数组进行操作。这种方法类似于数值数组的加法运算，但在字符串数组中，它被用于连接字符串元素。以下是一些示例，展示了如何在不同情况下使用加号(+)操作字符串数组：

s1 = ["李", "王", "张"]; s2 = ["一", "二", "三"]; s1 + s2	1×3 string 数组 "李一" "王二" "张三"
s1 + s2 % 支持算术运算的五种兼容模式	3×3 string 数组 "李一" "王二" "张三" "李二" "王二" "张三" "李三" "王二" "张三"
s1 + '清风' % 和字符串向量进行连接	1×3 string 数组 "李清风" "王清风" "张清风"
% 也能和字符串元胞数组进行连接 s1 + {'j','k','zq'}	1×3 string 数组 "李j" "王jk" "张zq"
s1 + [1,2,3] % 和数值数组进行连接	1×3 string 数组 "李1" "王2" "张3"
s1 + 1:3 % 易错点！	报错：'string' 类型的操作数不支持运算符 '+'。
s1 + (1:3) % 指定运算优先级	1×3 string 数组 "李1" "王2" "张3"
"指标" + (1:3)	1×3 string 数组 "指标1" "指标2" "指标3"

拼接操作：字符串数组的拼接方式和数值数组非常相似，我们既可以使用中括号[]进行拼接，还可以借助cat函数、horzcat函数和vertcat函数进行拼接。另外，拼接的文本类型除了字符串之外，也支持字符串元胞数组、字符串元胞数组、数值数组等，MATLAB会自动将它们转换成字符串类型，这一操作类似于我们前面介绍的混合创建字符串数组。

s1 = ["ab", "cd"]; s2 = ["ef", "gh"]; [s1,s2] % 等价于 horzcat(s1,s2) 或 cat(2,s1,s2) [s1;s2] % 等价于 vertcat(s1,s2) 或 cat(1,s1,s2)	1×4 string 数组 "ab" "cd" "ef" "gh"
[s1;66 88]	2×2 string 数组 "ab" "cd" "66" "88"
[s2 'ij']	1×3 string 数组 "ef" "gh" "ij"
[s1;s2;{'ij','kl'}]	3×2 string 数组 "ab" "cd" "ef" "gh" "ij" "kl"

重构操作：我们可以使用reshape函数对字符串数组进行重构操作。

s = string(1:12)	1×12 string 数组 "1" "2" ... "12"
reshape(s,3,4) % 等价于下面三种写法： % reshape(s,[,4]) % reshape(s,3,[,]) % reshape(s,[3,4])	3×4 string 数组 "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

重复操作：在第三章矩阵操作中，我们介绍过repmat函数和repelem函数，它们分别用于对数组整体进行堆叠和对数组中的元素进行重复。这两个函数也适用于字符串数组，我们来看例子：

s = ["a", "b"]; repmat(s,3,2)	3×4 string 数组 "a" "b" "a" "b" "a" "b" "a" "b" "a" "b" "a" "b"
repelem(s,2)	1×4 string 数组 "a" "a" "b" "b"
repelem(s,[2,3])	1×5 string 数组 "a" "a" "b" "b" "b"

排序操作：第三章中我们介绍过两个排序函数，分别是sort函数和sortrows函数。其中，sort函数通常用于对行进行排序，sortrows函数则基于矩阵的某一列对矩阵进行排序，排序后得到的新矩阵的同一行元素不会改变。

以升序排为例，字符串在排序时是逐字符进行比较的，对于每一对比较的字符，它们的Unicode编码被用来确定排序顺序，Unicode编码较小的字符会被排在前面。如果一个较短的字符串的所有字符和另一个更长的字符串的前面字符完全相同，那么这个较短的字符串排在前面。

s = ["aa","A","ab","z","ac","123","Bc","ba","bad","abc"]; [sort_s, ind] = sort(s)	
sort_s = 1×10 string 数组 "z" "123" "A" "Bc" "aa" "ab" "abc" "ac" "ba" "bad"	
ind = 4 6 2 7 1 3 10 5 8 9	

字符串数组的关系运算是基于字符的 Unicode 编码进行的。对于字符串之间的比较，MATLAB 会逐个字符地比较它们的 Unicode 编码，直到发现第一个不同的字符或达到其中一个字符串的末尾。

s = ["ab","cd","b"; "Ab","ce","bc"; "c","cd","def"]; s == "ab" % 等价于 strcmp(s,"ab")	3×3 logical 数组 1 0 0 0 0 0 0 0 0
strcmpi(s,"ab") % 不区分大小写	3×3 logical 数组 1 0 0 1 0 0 0 0 0
strncmp(s,"c",1) % 比较第一个字符	3×3 logical 数组 0 1 0 0 1 1 1 1 0
s == ["c","cd","ccc"] % 支持第三章介绍的算术运算中的五种兼容模式	3×3 logical 数组 0 1 0 0 0 0 1 1 0

s1 = ["Ac","ab","aa","Ab","a"]; s2 = ["Aa","Ab","aa"]; unique(s1)	1×5 string 数组 "Ab" "Ac" "a" "aa" "ab"
% 按照原来的顺序输出唯一值 unique(s1,'stable')	1×5 string 数组 "Ac" "ab" "aa" "Ab" "a"
ismember(s1, s2)	1×6 logical 数组 0 0 1 0 1 0
ismember(s2, s1)	1×3 logical 数组 0 1 1
intersect(s1, s2) % 交集	1×2 string 数组 "Ab"
union(s1, s2) % 并集	1×6 string 数组 "Aa" "Ab" "Ac" "a" "aa" "ab"
setdiff(s1, s2) % 差集	1×3 string 数组 "Ac" "a" "ab"
setxor(s1, s2) % 对称差集	1×4 string 数组 "Aa" "Ac" "a" "ab"

字符串数组的运算方法

(1) 关系运算

(2) 集合运算