

Matlab与C混合编程之二:C/C++ MEX API

zcl.space

目录

1 引言	2
2 matlab提供的数据类型	2
3 matlab提供的函数API	2
3.1 创建或者删除数组	2
3.2 验证数据	3
3.3 读取数据	3
3.4 数据类型转换	3

1 引言

为方便matlab和C混合编程，matlab定义了一些可以在C代码中使用的数据类型和函数。本文首先给出这些数据类型和函数，然后举例演示这些数据类型和函数的使用过程。由于各个API之间相对独立，在阅读本文的过程中，读者随时可以通过目录按钮跳转到自己感兴趣的内容。也可以通过留言板向我提问（我的主页地址是 `zcl.space`）。

2 matlab提供的数据类型

为了方便编写能够在matlab中调用的C代码（通过MEX文件调用），matlab定义了一些数据类型。在编写C代码过程中，程序员使用这些数据类型就像使用C语言的 `int char` 那样简单。表1给出了matlab提供的数据类型以及简单的描述。

表 1: matlab提供的数据类型

数据类型	描述
<code>mxArray</code>	matlab array, 是的在C中可以定义matlab array
<code>mwSize</code>	专门为描述matlab的矩阵大小定义的类型，主要是为了方便代码跨平台，大多数情况 <code>int</code> 也可以
<code>mxIndex</code>	为索引变量定义的类型，与 <code>mwSize</code> 差不多
<code>mwSignedIndex</code>	有符号整型，主要为描述矩阵大小定义
<code>mwChar</code>	为 <code>string</code> 数组定义的类型
<code>mxLogical</code>	为 <code>logical</code> 数组定义的类型
<code>mxClassID</code>	
<code>mxComplexity</code>	指示数组元素是否包含复数的旗标（flag）
<code>mxGetEps</code>	EPS
<code>mxGetInf</code>	无穷大
<code>mxGetNaN</code>	NaN

3 matlab提供的函数API

matlab为C编程提供了丰富的API。根据其功能，这些API主要可以分为以下四类：

1. 创建或者删除数组
2. 验证数据有效性
3. 存取数据
4. 数据类型转换

我们分四个小节简要列出这些函数名称及其功能。

3.1 创建或者删除数组

这一类API方便程序员在C代码中创建特定类型的数组，并为其分配或者释放内存。表2给出了这类API的名称和相关定义。

表 2: 创建删除数组API

名称	功能
<code>mxCreateDoubleMatrix</code>	创建二维双精度浮点数组
<code>mxCreateDoubleScale</code>	标量, 双精度数组 (初始化为特定值)
<code>mxCreateNumericMatrix</code>	创建二维数值矩阵
<code>mxCreateNumericArray</code>	创建N维数值数组
<code>mxCreateUninitNumericMatrix</code>	Uninitialized 2-D numeric matrix
<code>mxCreateUninintNumericArray</code>	未初始化的N维数组
<code>mxCreateString</code>	一维符号数组, 字符串
<code>mxCreateCharMatrixFromStrings</code>	二维字符串数组
<code>mxStringCharArray</code>	N维字符串数组
<code>mxCreateLogicalScalar</code>	标量或者逻辑变量数组
<code>mxCreatelogicalMatrix</code>	二维逻辑数组
<code>mxCreateLogicalArray</code>	N维逻辑数组
<code>mxCreateSparseLogicalMatrix</code>	二维稀疏逻辑数组
<code>mxCreateSparse</code>	二维稀疏数组
<code>mxCreateStructMatrix</code>	二维 structure数组
<code>mxCreateStructArray</code>	N维structure数组
<code>mxCreateCellMatrix</code>	二维cell数组
<code>mxCreateCellArray</code>	N维structure数组
<code>mxDestroyArray</code>	释放分配的内存空间
<code>mxDuplicateArray</code>	复制数组空间
<code>mxCalloc</code>	使用matlab的内存管理器分配数组, 初始化为0
<code>mxMalloc</code>	使用matlab的内存管理器分配未初始化的数组
<code>mxRealloc</code>	使用matlab的内存管理器再次动态分配内存
<code>mxFree</code>	释放使用matlab内存管理器分配的内存

3.2 验证数据

matlab为验证数据也提供了一些API, 这些API的主要作用是判断输入数据的类型。如表3所示。

3.3 读取数据

matlab为数据读写定义了一些API。这些API在 `mexFunction` 函数中使用非常广泛, 极大的方便了matlab和C之间的数据传递。表4给出了这些函数名称和功能简述

3.4 数据类型转换

matlab为字符和数组之间的数据类型转换提供了API, 如表tab:20151111ConvertDataTypes所示。

表 3: 验证数据API

名称	功能
<code>mxIsDoulbe</code>	判断mxArray是不是双精度浮点数
<code>mxIsSingle</code>	判断mxArray是不是单精度浮点数
<code>mxIsComplex</code>	判断是否为复数
<code>mxIsNumeric</code>	判断是否为numeric
<code>mxIsInt64</code>	判断是不是64bit整型
<code>mxIsUInt64</code>	判断是不是无符号64bit整型
<code>mxIsInt32</code>	判断是不是32位整型
<code>mxIsUInt32</code>	判断是不是无符号32位整型
<code>mxIsInt16</code>	判断是不是16位整型
<code>mxIsUInt16</code>	判断是不是无符号16位整型
<code>mxIsInt8</code>	判断是不是16位整型
<code>mxIsUInt8</code>	判断是不是无符号16位整型
<code>mxIsScalar</code>	判断是不是scalar数组
<code>mxIsChar</code>	判断是不是符号数组
<code>mxIsLogical</code>	判断是不是逻辑
<code>mxIsLogicalScalar</code>	判断是不是逻辑数组
<code>mxIsLogicalScalarTrue</code>	判断逻辑数组是不是真
<code>mxIsStruct</code>	判断是否为sturcture数组
<code>mxIsCell</code>	判断是不是cell数组
<code>mxIsClass</code>	判断是不是特定的类型
<code>mxIsInf</code>	判断输入是不是无穷大
<code>mxIsFinite</code>	判断输入是不是有限大
<code>mxIsNaN</code>	判断输入是不是NaN
<code>mxIsEmpty</code>	判断数组是否为空
<code>mxIsSparse</code>	判断输入是否为稀疏数组
<code>mxIsFromGlobalWS</code>	判断数组是否从matlab全局空间传入
<code>mxAssert</code>	判断assertion值（用于debug）
<code>mxAssertS</code>	判断assertion值，不打印assertion文本

表 4: 读取数据API

名称	功能简述
<code>mxGetNumberOfDimensions</code>	获取数组的维度
<code>mxGetElementSize</code>	每一个数据元素需要的字节数
<code>mxGetDimensions</code>	指向数组维度的指针
<code>mxSetDimensions</code>	修改维度和每一维的大小
<code>mxGetNumberOfElements</code>	数组中元素的个数
<code>mxCalcSingleSubscript</code>	相对于起始元素的偏移量
<code>mxGetM</code>	数组元素的行数
<code>mxGetN</code>	数组元素的列数
<code>mxSetM</code>	设置数组元素的行数
<code>mxSetN</code>	设置数组元素的列数
<code>mxGetScalars</code>	第一个数据元素的实部
<code>mxGetPr</code>	数组的实部
<code>mxSetPr</code>	设置数组的实部
<code>mxGetPi</code>	数组的虚部
<code>mxSetPi</code>	设置数组的虚部
<code>mxGetData</code>	指向数组中实部的指针
<code>mxSetData</code>	设置指向数组中实部的指针
<code>mxGetImagData</code>	指向数组中虚部的指针
<code>mxSetImagData</code>	设置指向数组中虚部的指针
<code>mxGetChars</code>	指向字符数组的指针
<code>mxGetLogicals</code>	指向逻辑类型数组的指针
<code>mxGetClassID</code>	数组的类型
<code>mxGetClassName</code>	数组的类型（以字符串返回）
<code>mxSetClassName</code>	把C中的Array指定为matlab中的array
<code>mxGetProperty</code>	matlab对象的公共值
<code>mxSetProperty</code>	设置matlab对象的公共值
<code>mxGetField</code>	获取structure的域（给定index和name）
<code>mxSetField</code>	设置structure的域
<code>mxGetNumberOfFields</code>	structure中的域个数
<code>mxGetFieldNameByNumber</code>	给定域编号获得域的名字
<code>mxGetFieldNumber</code>	给定域名字获得域编号
<code>mxGetFieldByNumber</code>	给定域索引和域数值获得域值
<code>mxSetFieldByNumber</code>	给定索引和域数值，设置域值
<code>mxAddField</code>	在结构体中添加域
<code>mxRemoveField</code>	在结构体中去除域
<code>mxGetCell</code>	获取Cell数组的值
<code>mxSetCell</code>	设置Cell数组的值
<code>mxGetNzmax</code>	IR, PR, PI数组的元素个数
<code>mxSetNzmax</code>	为非零元素设置空间数
<code>mxGetIr</code>	稀疏IR数组
<code>mxSetIr</code>	稀疏数组的IR数组
<code>mxGetJc</code>	稀疏JC数组
<code>mxSetJc</code>	JC数组的稀疏数组

表 5: 数据类型转换

名称	功能简述
<code>mxArrayToString</code>	array向字符串转化
<code>mxArrayToUTF8String</code>	array向字符串转化（用UTF8编码）
<code>mxGetString</code>	把string数组转化为C-style字符串
<code>mxSetClassName</code>	结构体数组转向matlab对象数组