
开发说明书

1 开发环境.....	1
2 配置说明.....	1
2.1 MATLAB R2012a 配置过程.....	1
3 工程文件说明.....	5
4 关键代码说明.....	7
4.1 代码处理过程说明.....	7
4.2 类说明.....	8
4.3 主要函数说明.....	8

1 开发环境

Visual Stdlio 2008, C++, Matlab R2012a

2 配置说明

2.1 MATLAB R2012a 配置过程

(安装路径: D:\Program Files\MATLAB\R2012a)

在 matlab 中选择 compiler

```
>> mbuild -setup

Welcome to mbuild -setup. This utility will help you set up
a default compiler. For a list of supported compilers, see
http://www.mathworks.com/support/compilers/R2012a/win64.html

Please choose your compiler for building shared libraries or COM components:

Would you like mbuild to locate installed compilers [y]/n? y

Select a compiler:
[1] Microsoft Visual C++ 2010 in C:\Program Files (x86)\Microsoft Visual Studio 10.0\
[2] Microsoft Visual C++ 2008 SP1 in C:\Program Files (x86)\Microsoft Visual Studio 9.0

[0] None

Compiler: 2

Please verify your choices:

Compiler: Microsoft Visual C++ 2008 SP1
Location: C:\Program Files (x86)\Microsoft Visual Studio 9.0

Are these correct [y]/n? y
```

```
*****
Warning: Applications/components generated using Microsoft Visual Studio
2008 require that the Microsoft Visual Studio 2008 run-time
libraries be available on the computer used for deployment.
To redistribute your applications/components, be sure that the
deployment machine has these run-time libraries.
*****

Trying to update options file: C:\Users\goitsys\AppData\Roaming\MathWorks\MATLAB\R2012a\compopts.bat
From template:          D:\MATLAB\R2012a\bin\win64\mbuildopts\msvc90comp.bat

Done . . .

x >>
```

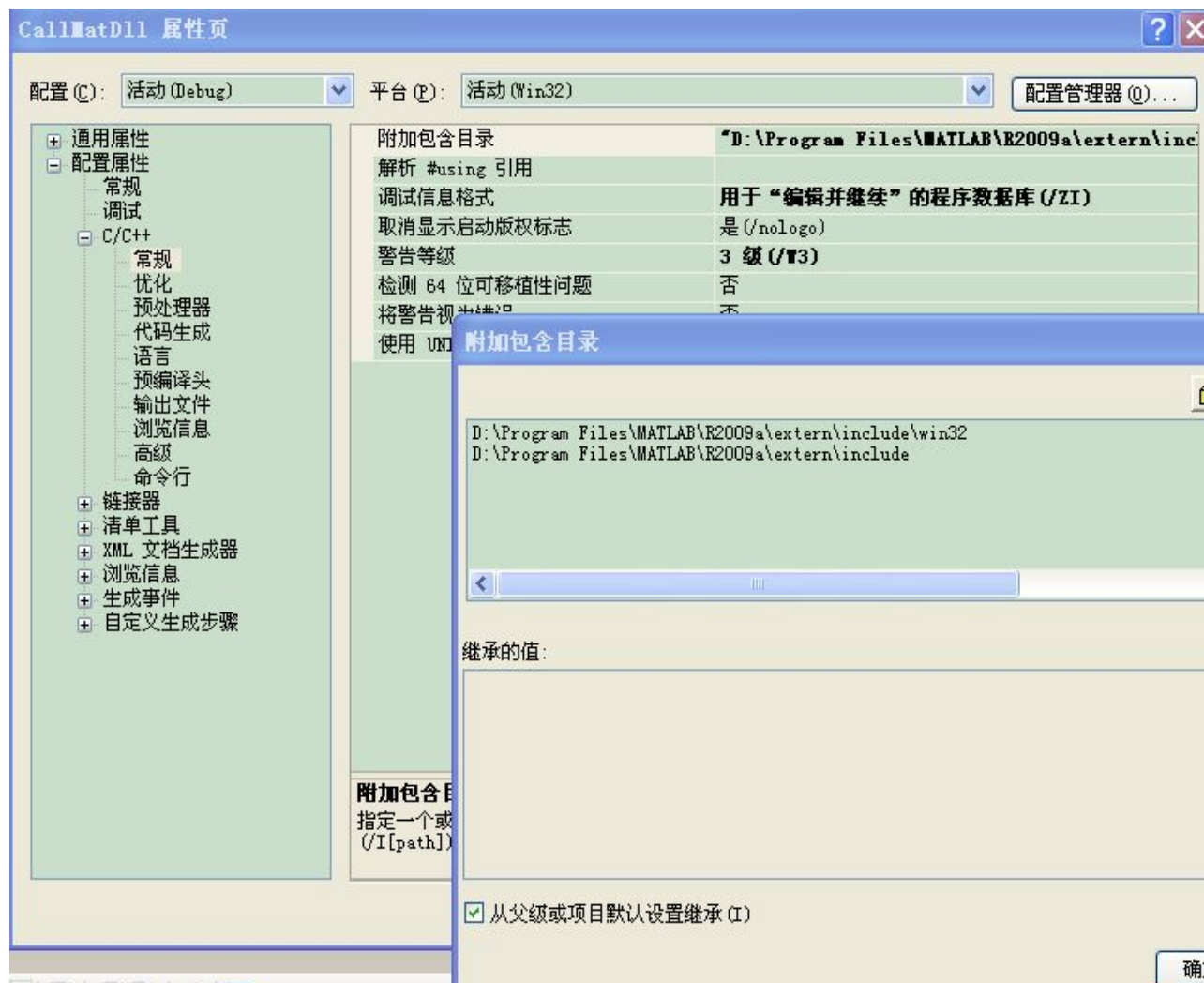
2.2 VS2008 中文版配置过程

（安装路径： "C:\Program Files (x86)\Microsoft Visual Studio 9.0）

新建一个 win32 控制台应用程序，取名 util_matlab。接下来进行如下配置。

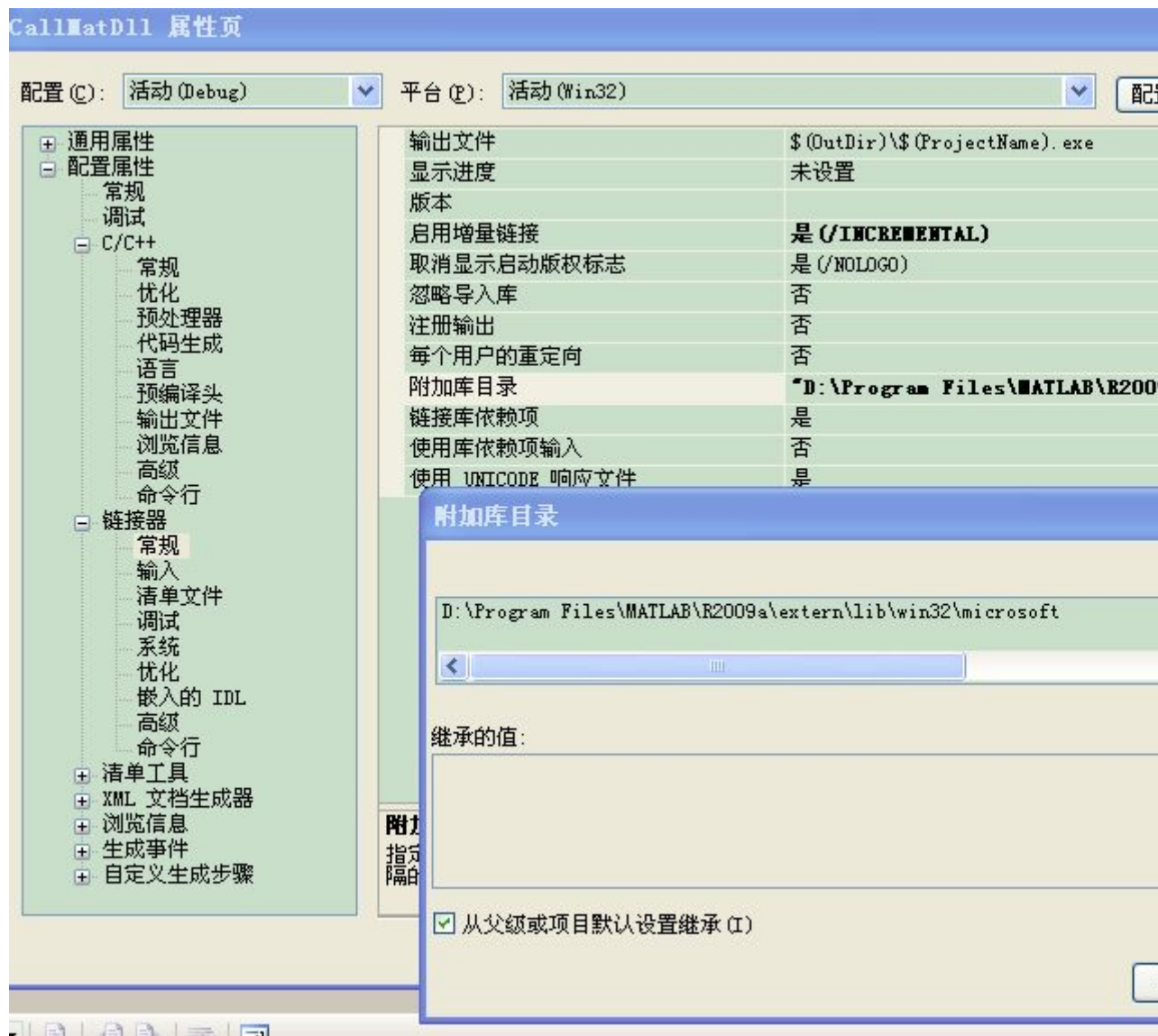
解决方案->右键->属性->配置属性->C/C++->常规->附加包含目录

配置内容如下：



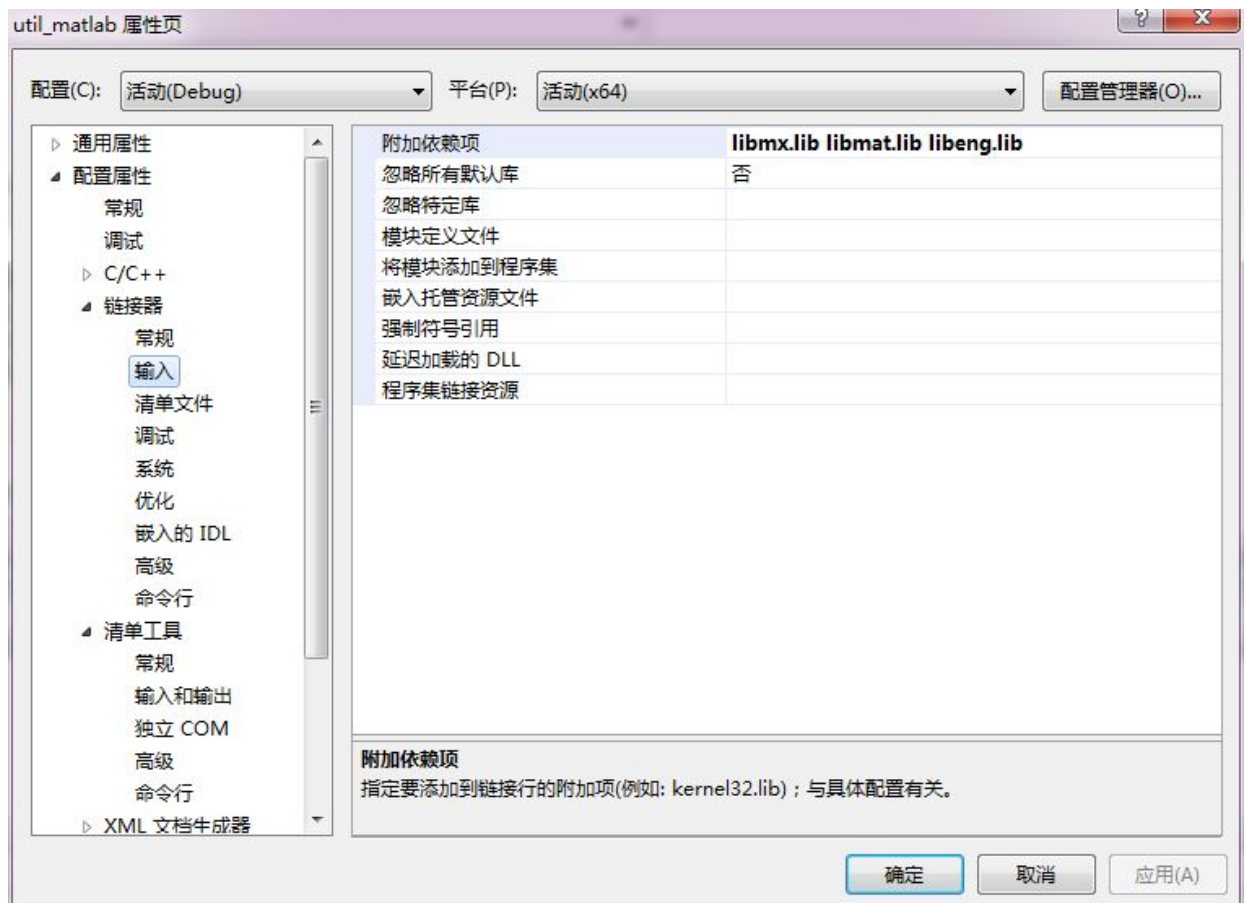
配置项目属性页/配置属性/链接器/常规/附加库目录

配置内容如下：



配置项目属性页/配置属性/链接器/输入/附加依赖性

加入 libmx.lib, libmat.lib, libeng.lib 三项, 如下:



2.3 系统环境变量配置

将系统环境变量 Path 配置成如下值:

D:\MATLAB\R2012a\runtime\win64;D:\MATLAB\R2012a\bin\win64

2.4 参考资料

<http://blog.csdn.net/infocarrier/article/details/5854522>

<http://blog.csdn.net/hahalxp/article/details/5415257>

<http://www.cnblogs.com/EverYoung/archive/2012/03/29/2423418.html>

3 工程文件说明

解决方案名: util_matlab



头文件\util_space.h：测试类 util_space 的头文件，包含所有类成员的声明；

```
util_space(void);
~util_space(void);
void isempty(double *,int);
void hist(double *,int);
char * adline(double *,double *,double *,double *,int ,bool);
char * pcpval(void);
char * frontconn(double **array,char*a,char* numports);
void cumsum(int *,int);
char * dayed(void);
char * todayed(void);
char *datefinded( char date[20],char superSet[20]);
```

源文件\util_space.cpp：测试类 util_space 的源文件，包含所有函数的定义及处理；

源文件\interface.cpp：测试类 util_space 的应用程序入口。

4 关键代码说明

4.1 代码处理过程说明

```
char* util_space::adline(double * highp, double * lowp, double * closep, double * tvolume, int size, bool plot)
{
    char *result=new char[2000]; //存储输出的结果

    engEvalString(ep, "clear"); //向Matlab发送清屏信息 → c1

    mxArray *highpmx = mxCreateDoubleMatrix(size, 1, mxREAL); //创建数组
    mxArray *lowpmx = mxCreateDoubleMatrix(size, 1, mxREAL);
    mxArray *closepmx = mxCreateDoubleMatrix(size, 1, mxREAL);
    mxArray *tvolumemx = mxCreateDoubleMatrix(size, 1, mxREAL); → c2

    memcpy(mxGetPr(highpmx), highp, size*sizeof(double)); //将数组highp复制到mxarray数组highpmx中
    memcpy(mxGetPr(lowpmx), lowp, size*sizeof(double)); //将数组lowp复制到mxarray数组lowp中 → c3
    memcpy(mxGetPr(closepmx), closep, size*sizeof(double)); //将数组closep复制到mxarray数组closep中
    memcpy(mxGetPr(tvolumemx), tvolume, size*sizeof(double)); //将数组tvolume复制到mxarray数组tvolumemx中

    engPutVariable(ep, "highp", highpmx); //将mxArray数组highpmx写入到Matlab工作空间, 命名为highp
    engPutVariable(ep, "lowp", lowpmx); //将mxArray数组lowpmx写入到Matlab工作空间, 命名为lowp → c4
    engPutVariable(ep, "closep", closepmx); //将mxArray数组closepmx写入到Matlab工作空间, 命名为closep
    engPutVariable(ep, "tvolume", tvolumemx); //将mxArray数组tvolumemx写入到Matlab工作空间, 命名为tvolume

    engOutputBuffer(ep, result, 2000); //开启控制台信息读取 → c5

    engEvalString(ep, "highp");
    engOutputBuffer(ep, NULL, 0); //结束matlab控制台信息读取
    cout<<result<<endl;
    engOutputBuffer(ep, result, 2000); //开启控制台信息读取
    engEvalString(ep, "lowp");
    engOutputBuffer(ep, NULL, 0); //结束matlab控制台信息读取
    cout<<result<<endl;
    engOutputBuffer(ep, result, 2000); //开启控制台信息读取
    engEvalString(ep, "closep");
    engOutputBuffer(ep, NULL, 0); //结束matlab控制台信息读取
    cout<<result<<endl;
    engOutputBuffer(ep, result, 2000); //开启控制台信息读取
    engEvalString(ep, "tvolume");
    engOutputBuffer(ep, NULL, 0); //结束matlab控制台信息读取
    cout<<result<<endl;
    engOutputBuffer(ep, result, 2000); //开启控制台信息读取
    engEvalString(ep, "adln=adline(highp, lowp, closep, tvolume)");
    engOutputBuffer(ep, NULL, 0); //结束matlab控制台信息读取 → c7
    cout<<result<<endl; → c8

    if(plot)
        engEvalString(ep, "plot(adln)"); → c9
    mxDestroyArray(highpmx); //销毁mxArray数组
    mxDestroyArray(lowpmx); //销毁mxArray数组
    mxDestroyArray(closepmx); //销毁mxArray数组
    mxDestroyArray(tvolumemx); //销毁mxArray数组 → c10

    return result; → c11
}
```

c1:清除 Matlab 之前存储的变量

c2:创建向量(数组), 用于存储数据

c3:将函数形参传递的数据复制到 mxArray 型数组中

c4:将 mxArray 型数组数据传入 Matlab, 并命名

c5:开启 Matlab 控制台信息读取, 并保存到 result 数组中

-
- c6:调用 Matlab 函数库进行具体的处理
 - c7:结束 Matlab 控制台信息的读取
 - c8:输出缓冲区 result 中保存的 Matlab 控制台的处理结果
 - c9:根据形参确定是否要调用 matlab 进行绘图
 - c10 销毁 mxArray 型数组，释放空间
 - c11:返回处理结果

4.2 类说明

```
class util_space
{
private:
    Engine *ep; //定义Matlab引擎指针
public:

    util_space(void);
    ~util_space(void);

    char * adline(double *, double *, double *, double *, int , bool);
    void isempty(double *, int);
    void hist(double *, int);
    char * pcval(void);
    char * frontconn(double **array, char*a, char* numports);
    void cumsum(int *, int);
    char * nowed(int i = 0);
    char * dayed(void);
    char * todayed(void);
    char *datefinded( char date[20], char superSet[20]);

};
```

4.3 主要函数说明

F1:char* adline(double * highp, double * lowp, double * closep, double *tvolume, int size, bool plot);

/**计算离散指标AD**/

/**Matlab函数原型: adln = adline(highp, lowp, closep, tvolume)**//

/**输入:

/**highp:double数组, 存放离散指标AD的输入数据序列highprice

/**lowp:double数组, 存放离散指标AD的输入数据序列lowprice

```
/**close:double数组，存放离散指标AD的输入数据序列closeprice  
/**tvolume:double数组，存放离散指标AD的输入数据序列tvolume  
/**size:整型变量，标识上述各个数组的长度  
/**plot:布尔变量，标识该处理结果是否需要通过matlab做出图形进行显示  
/**返回值:  
/**char*:将matlab执行的结果作为字符数组返回
```

```
F2:void isempty(double *,int);  
/**判断序列是否为空**//  
/**Matlab函数原型为: tf = isempty(fts)**//  
/**输入:  
/**objArray:double型数组，存放序列  
/**size:整型变量，标示数组的长度  
/**返回值:  
/**序列为空返回，否则返回
```

```
F3:void hist(double *,int);  
/**计算和显示数据的直方图tsobj系列金融序列中包含的对象**//  
/**Matlab函数原型为: hist(tsobj, numbins)**//  
/**输入:  
/**objArray:double型数组，存放序列  
/**size:整型变量，标示数组的长度  
/**返回值:  
/**在Matlab控制台输出直方图信息
```

```
F4:void cumsum(int *,int);  
/**计算每个序列数据系列的总合在金融序列对象oldfts并返回**//  
/**Matlab函数原型为: newfts = cumsum(oldfts)**//  
/**输入:  
/**objArray:int型数组，存放时间序列  
/**size:整型变量，标示数组的长度  
/**返回值:  
/**在Matlab控制台输出图型信息
```

```
F5:char* dayed(char str[100] );  
/**返回某一天是本月的第几天**//  
/**matlab原型:DayMonth = day(Date)**//  
/**输入:  
/**Date:char数组，存放你要测试的日期型如: mm/dd/yy**//  
/**返回值:  
/**char*, 将MATLAB执行的结果作为字符数组返回。
```

```
F6:char* todayed();  
/**返回当前的日期**//
```

```

/**matlab函数原型: Datenum = today**//
/**输入:
/**没有输入, 调用函数直接返回系统当前的日期**//
/**返回值:
/**char*将MATLAB执行的结果作为字符数组返回**//

```

```

F7:char* nowed(int i );
/**返回系统的当前时间**//
/**matlab函数原型:t = now t2 = rem(now,1)**//
/**输入:
/**整形i = 0表示需要返回系统的时间和日期**//
/**整形i = 1表示仅仅返回时间**//
/**返回值:
/**char*将MATLAB执行的结果作为字符数组返回

```

```

F8:char* datefinded( char date[20],char superSet[20]);
/**计算子集合日期在超集合中的前面元素和后面元素**//
/**matlab原型: Indices = datefind(Subset, Superset, Tolerance)**//
/**输入:
/**Subset:char数组, 存放子集日期的序列。**//
/**Superset:char数组, 存放超集的日期序列。**//
/**Tolerance:整形数, 默认值为。**//
/**返回值:
/**char*将 MATLAB 执行的结果作为字符数组返回。

```

```

F9:char* pcpval();
/**计算线性不等式解决总投资组合的价值。**//
/**Matlab函数原型: [A,b] = pcpval(PortValue, NumAssets)**//
/**输入:

/**PortValue:标量总资产组合的价值.
/**NumAssets:数量的可用资产投资.
/**返回值:

/**char*:将matlab执行的结果作为字符数组返回.

```

```

F10:char* frontconn(double **arry, char*a, char* numports);

/**计算均值-方差有效边界**//
/** Matlab函数原型:[PortRisk, PortReturn, PortWts] = frontcon(ExpReturn,
ExpCovariance, NumPorts, PortReturn, AssetBounds, Groups, GroupBounds,
varargin).
/**输入:
/** ExpReturn: 资产的数量(NASSETS) 向量指定每个资产的预期(平均值) 返回.

```

/**ExpCovariance: NASSETS-by-NASSETS指定资产收益的协方差矩阵.

/**NumPorts: (生成可选)数量的投资组合有效边界。回报之间的等距的最大。可能回归和最小的风险点。如果NumPorts空(进入[], frontcon计算等距的点。当进入一个目标回报率(PortReturn), 输入NumPorts空矩阵[]。

/**返回值:

/**char*: 将matlab执行的结果作为字符数组返回.

/**double: 将 matlab 执行的结果作为精度浮点型数组返回。