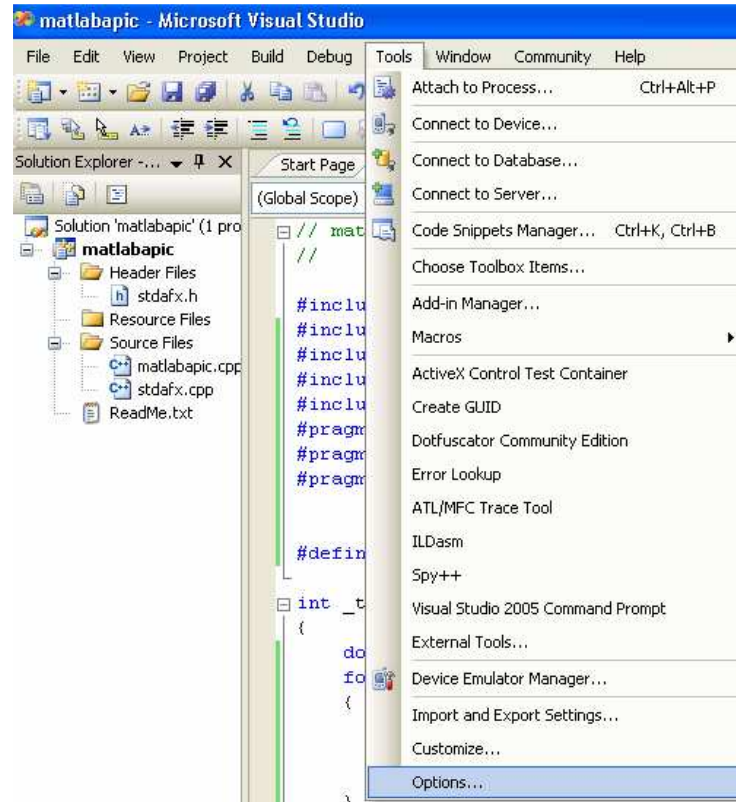


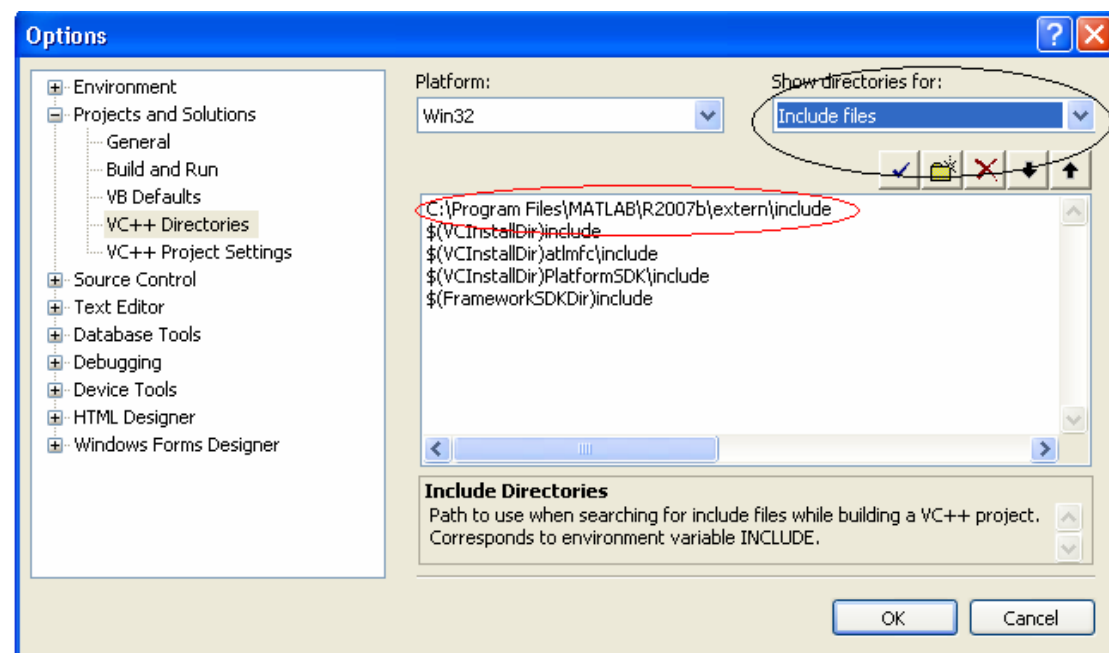
Matlab API (R2007b) in Microsoft Visual Studio.NET 2005

By : Sunu Wibirama

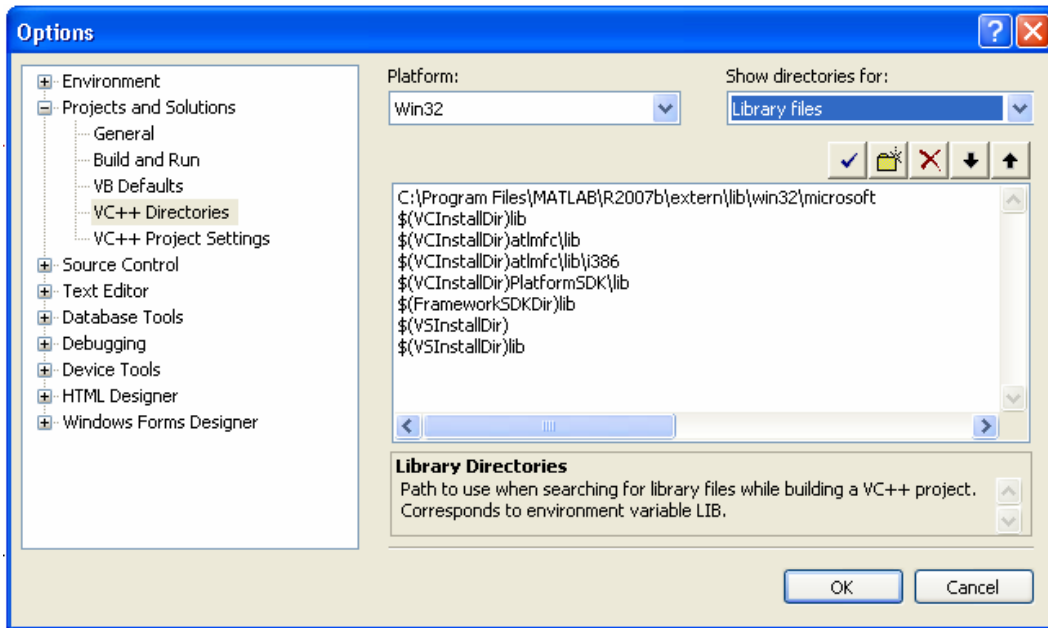
1. Open MSVS 2005 and create console application. Press Tools → Options to setup directory for Matlab API.



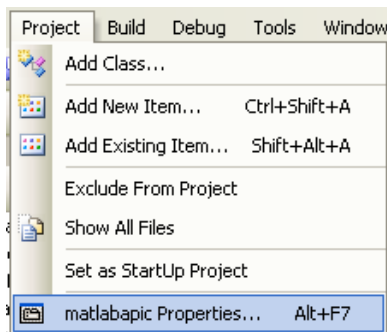
2. Select VC++ Directories. From the dropdown menu, select "Include files". Enter your Matlab path as shown in this pictures.



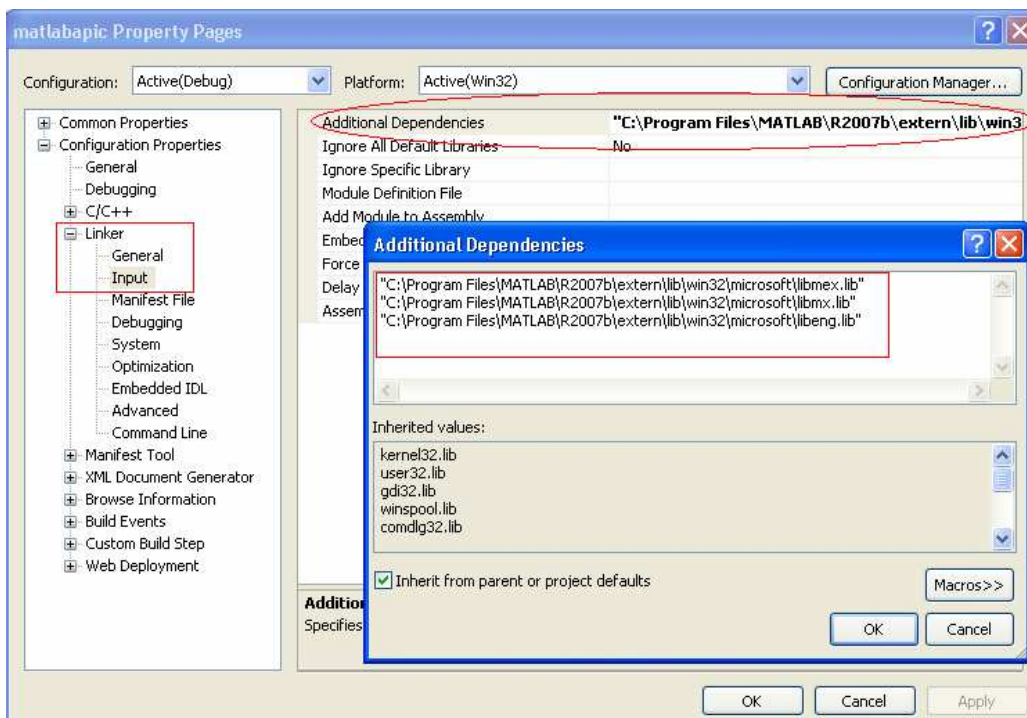
3. Still at the same windows, select "Library Files" from the drop down menu. Enter the Matlab library as shown in the picture below.



4. Select Project → Properties, as shown in the picture below



5. Enter some matlab library, as shown in the picture below



6. Enter this source code to your cpp file to try the Matlab API

```
#include "stdafx.h"
#include <stdio.h>
#include <conio.h>
#include <memory.h>
#include <engine.h> //Located in $Matlab$\extern\include
#pragma comment( lib, "Libmx.lib" )
#pragma comment( lib, "libmex.lib" )
#pragma comment( lib, "libeng.lib" )
#define SIZE 50

int _tmain(int argc, _TCHAR* argv[])
{
    double x[SIZE], y1[SIZE], y2[SIZE];
    for (int i=0; i<SIZE; i++)
    {
        x[i]=((double)i)*0.1;
        y1[i]=x[i]*x[i];
        y2[i]=y1[i]*x[i];
    }

    Engine *m_pEngine;
    m_pEngine = engOpen(NULL);
    if (m_pEngine == NULL)
    {
        //Error! Fail to connect to MATLAB engine.
        // The plot function will be disabled!
        printf("Fail to open MATLAB Engine!\n");
        exit(0);
    }

    //Translate data from C++ to Matlab
    mxArray *m_X, *m_Y1, *m_Y2;
    m_X=mxCreateDoubleMatrix(1, SIZE, mxREAL);
    memcpy((void *)mxGetPr(m_X), (void *)x, sizeof(double)*SIZE);
    engPutVariable(m_pEngine, "x", m_X);

    m_Y1=mxCreateDoubleMatrix(1, SIZE, mxREAL);
    memcpy((void *)mxGetPr(m_Y1), (void *)y1, sizeof(double)*SIZE);
    engPutVariable(m_pEngine, "y1", m_Y1);

    m_Y2=mxCreateDoubleMatrix(1, SIZE, mxREAL);
    memcpy((void *)mxGetPr(m_Y2), (void *)y2, sizeof(double)*SIZE);
    engPutVariable(m_pEngine, "y2", m_Y2);

    //Plot by sending command to engine
    engEvalString(m_pEngine, "plot(x, y1, '- ', x, y2, ':');");
    engEvalString(m_pEngine, "title('y_1/y_2 vs. x')");
    engEvalString(m_pEngine, "xlabel('x');");
    engEvalString(m_pEngine, "ylabel('y (Unit:10)');");

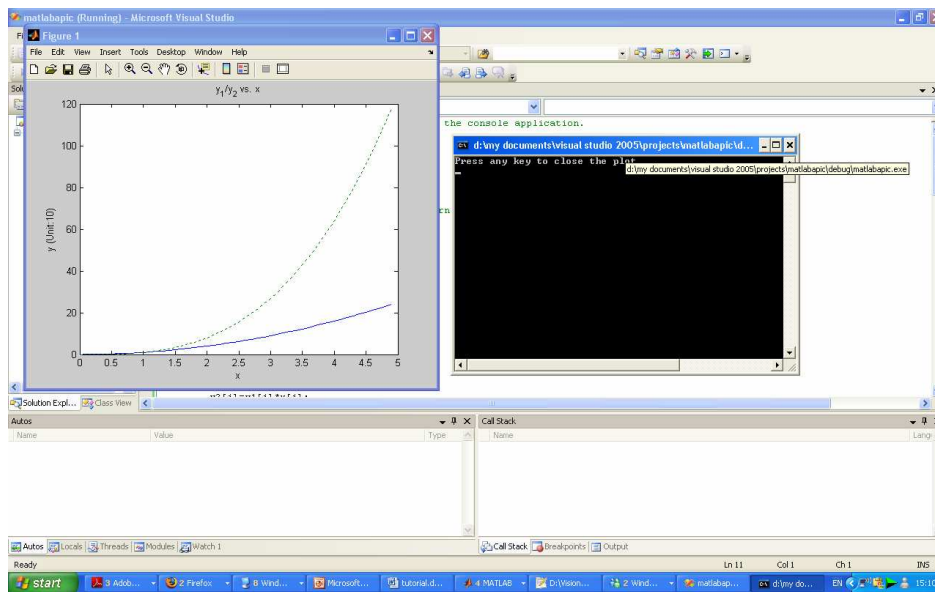
    printf("Press any key to close the plot...\n");
    getch();

    //Close plot window
    engEvalString(m_pEngine, "close;");

    engClose(m_pEngine);

    printf("Press any key to exit...\n");
    getch();
    return 0;
}
```

RESULTS :



References :

1. <http://www.codeproject.com/KB/DLL/MatlabSharedLib.aspx?fid=30854&df=90&mpp=25&noise=3&sort=Position&view=Quick&select=2281650>
2. Matlab API in C++ (Presentation file by : Christopher Dabney), accessed from www.cse.ucsc.edu/classes/cmps060m/Spring06/ma/cdabney19/MatLab%20API%20to%20C++.ppt