

## C++调用 Fortran 函数

**C/C++编译器：MSVS2010（VC9），对 VS2008 也适用。**

**Fortran 编译器：Intel Visual Fortran Compiler 11.0，IVF 的其它版本也适用。**

采用动态库 **dll** 链接的方式。

一、创建 **FORTRAN DLL** 工程，生成 **forsubs.dll** 文件供调用。

□ 用IVF建立一个Dynamic-link library项目，比如项目名为forsubs.dll。在项目中新建f90代码文件，比如文件名为forsubs.f90。在forsubs.f90中写下如下代码：

```
! forsubs.f90
!
!  FUNCTIONS/SUBROUTINES exported from FORSUBS.dll:
! FORSUBS      - subroutine
!
INTEGER*4 FUNCTION Fact (n)
  !DEC$ ATTRIBUTES DLLEXPORT::Fact
  INTEGER*4 n [VALUE]
  INTEGER*4 i, amt
  amt = 1
  DO i = 1, n
    amt = amt * i
  END DO
  Fact = amt
  write(*,*)"Mixed calls succeed!"
END

SUBROUTINE Pythagoras (a, b, c)
  !DEC$ ATTRIBUTES DLLEXPORT::Pythagoras
  REAL*4 a [VALUE]
  REAL*4 b [VALUE]
  REAL*4 c [REFERENCE]
  c = SQRT (a * a + b * b)
END
```

编译后，你会找到FortranDLL.lib 和FortranDLL.dll，它们是后面调用所需要的。因为建立的是动态库项目，所以不会有.exe 生成。

二、创建**win32 console application**，调用**forsubs.dll**。

□ 用MSVS2010建立一个C++的Win32控制台应用程序，当然MFC的也可以。代码如下：

```
#include "stdafx.h"
#include <stdio.h>
#include <iostream>
#include <windows.h>

using namespace std;

int main()
```

```

{
//声明调用约定
typedef int (__cdecl * FACT)(int n);
typedef void (__cdecl * PYTHAGORAS)(float a, float b, float *c);
//加载动态库文件
HINSTANCE hLibrary=LoadLibrary(L"forsubs.dll");
if(hLibrary==NULL)
{
    cout<<"can't find the dll file"<<endl;
    return -1;
}

//获得Fortran导出函数FACT的地址
FACT fact=(FACT)GetProcAddress(hLibrary,"FACT");
if(fact==NULL)
{
    cout<<"can't find the function file."<<endl;
    return -2;
}

//获得Fortran导出函数PYTHAGORAS的地址
PYTHAGORAS pythagoras=(PYTHAGORAS)GetProcAddress(hLibrary,"PYTHAGORAS");
if(pythagoras==NULL)
{
    cout<<"can't find the function file."<<endl;
    return -2;
}

    float c;
    printf("Factorial of 7 is: %d\n", fact(7));
    pythagoras (30, 40, &c);
    printf("Hypotenuse if sides 30, 40 is: %f\n", c);
FreeLibrary(hLibrary); //卸载动态库文件
return 0;

}

```

- ☐ 将之前生成的forsubs.lib连同路径一起添加在：项目->属性->属性配置->连接器->输入：附加依赖项里。如果它在当前目录下，就填forsubs.lib就可以了。
  - ☐ 将之前生成的forsubs.dll放在当前目录下，在执行时程序要连接它的。当然，也可以把它放在任意目录下，然后通过系统的“环境变量”中的Path来指定这个目录。
- 编译生成，运行C++即可