**实验12-2 lua调用C模块**

**实验目的**

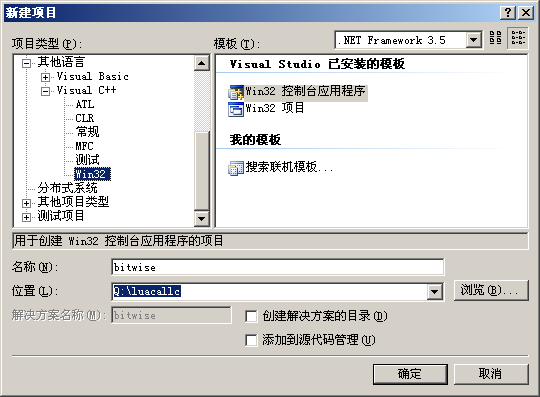
（1）理解lua 调用的C模块的原理；

（2）掌握C模块的编写方法；

（3）掌握lua调用C模块的方法。

**实验内容**

（1）打开vs2010，创建一个DLL项目(注意项目属性中项目所依赖的包含文件和库文件的目录必须添加进去；链接器中必须添加上附加依赖项lua.lib)：





（2）将以下代码复制到 bitwise.cpp 文件中

#include "stdafx.h"

#include <lua.hpp>

int l\_XOR(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-2);

int y = lua\_tointeger(L,-1);

z = x ^ y;

lua\_pushinteger(L,z);

return 1;

}

int l\_AND(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-2);

int y = lua\_tointeger(L,-1);

z = x & y;

lua\_pushinteger(L,z);

return 1;

}

int l\_OR(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-2);

int y = lua\_tointeger(L,-1);

z = x | y;

lua\_pushinteger(L,z);

return 1;

}

int l\_NOT(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-1);

z = ~x ;

lua\_pushinteger(L,z);

return 1;

}

int l\_LSHIFT(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-2);

int y = lua\_tointeger(L,-1);

z = x << y;

lua\_pushinteger(L,z);

return 1;

}

int l\_RSHIFT(lua\_State \* L)

{

int z;

int x = lua\_tointeger(L,-2);

int y = lua\_tointeger(L,-1);

z = x >> y;

lua\_pushinteger(L,z);

return 1;

}

int luaopen\_bitwise(lua\_State \* L)

{

luaL\_Reg mylib[] = {

{"AND",l\_AND},

{"OR",l\_OR},

{"XOR",l\_XOR},

{"NOT",l\_NOT},

{"LSHIFT",l\_LSHIFT},

{"RSHIFT",l\_RSHIFT},

{NULL,NULL},

};

luaL\_register(L,"bitwise",mylib);

return 1;

}

/\* 使用luaL\_newlib替代luaL\_register。

int luaopen\_bitwise (lua\_State \*L)

{

lua\_Reg mylib [] = {

{"AND",l\_AND},

{"OR",l\_OR},

{"XOR",l\_XOR},

{"NOT",l\_NOT},

{"LSHIFT",l\_LSHIFT},

{"RSHIFT",l\_RSHIFT},

{NULL,NULL},

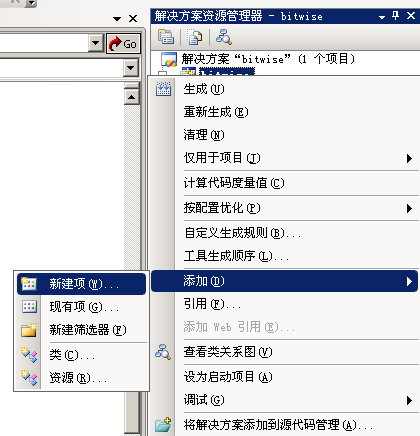
};

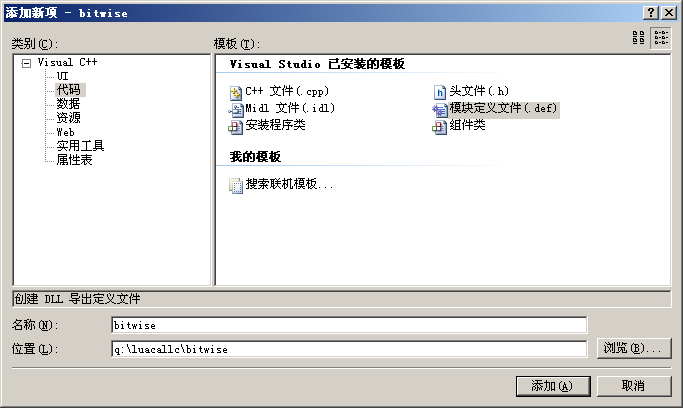
luaL\_newlib(L, mylib);

return 1;

}\*/

（3）添加模块定义文件，以导出函数





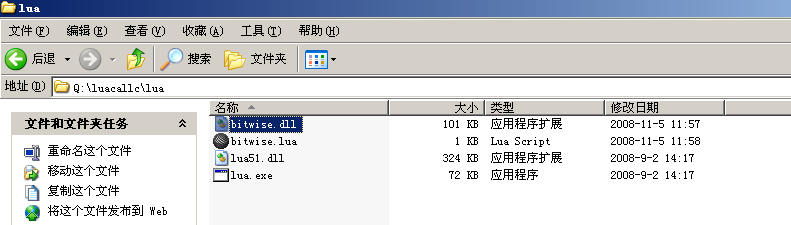
以下为 bitwise.def 文件的代码

LIBRARY "bitwise"

EXPORTS

luaopen\_bitwise

（4）生成项目，得到 bitwise.dll，我们将它复制到 lua 目录中



（5）打开记事本，写以下语句,保存到 lua 目录中，文件名为 callBitwise.lua。

**Lua提供高级的require函数来加载运行库，lua中的require函数功能主要有：**

**1）require函数搜索目录加载文件**

**2）require会判断是否文件已经加载避免重复加载同一文件。**

require "bitwise"

print(bitwise.AND(1,128))

print(bitwise.OR(1,128))

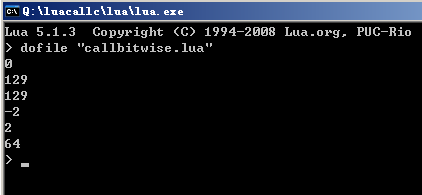
print(bitwise.XOR(1,128))

print(bitwise.NOT(1))

print(bitwise.LSHIFT(1,1))

print(bitwise.RSHIFT(128,1))

（6）打开 lua 解释器，运行 callBitwise.lua 脚本



（7）根据以上步骤修改C模块，添加求最大值的函数，使之返回两个数中最大值，并在lua脚本中调用该C模块中的求最大值函数。

（8）参考第7题中的求最大值函数，添加一个求最大值和最小值的函数，该函数返回两个值，一个最大值，一个最小值，并在lua文件中调用该函数。

**实验要求**：

（1）独立完成；

（2）win7以上兼容系统，lua解释器，VS2010以上版本；

（3）记录并分析实验结果。

**实验报告要求**：

（1）按照学院的实验报告格式编写实验报告(注意有封面的)，不能直接使用本指导书修改；

（2）指导书截图不用复制到实验报告，实验报告要附上自己的关键代码。最主要是要有合理的实验分析和总结；

（3）以“学号+姓名+实验\*”命名word文档，提交到ftp对应位置。