

深圳大学实验报告

课程名称： 中间件技术

实验项目名称： 实验 4 应用 JNI 跨语言混合编程

学院： 计算机与软件学院

专业： 软件工程

指导教师： 毛斐巧

报告人： 郑彦薇 学号： 2020151022 班级： 软件工程 01 班

实验时间： 2022 年 11 月 29 日、12 月 13 日（周二）

实验报告提交时间： 2022/12/6

教务部制

一、实验目的与要求:

通过编程问题的解决，直观的感受、体会和学用 Java JNI 解决异构语言编程问题，在 Java 程序中调用 C++ 程序。

https://blog.csdn.net/qq_41969790/article/details/106083780?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.no_search_link&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.no_search_link

二、实验内容

内容：解决以下异构语言编程互操作问题

在某软件开发项目中需要混合编程，其中，算法程序使用 C/C++ 编写，而算法服务器是用 Java 编写，使用 Java JNI 来实现 C/C++ 算法部分代码的调用。

编程过程分如下几步：

1.创建 Java 工程

- (1) 新建 Java 工程
- (2) 编写对应 C/C++ 程序的 Java 对象类

2.编译制作 dll 动态库

- (1) 创建 dll 工程
- (2) 编写 C/C++ 源文件
- (3) 生成 DLL 动态链接库文件

3.Java 程序调用

详细编程过程指导，请参考实验 4 应用 JNI 实现 Java 调用 C/C++ 代码编程指导.PDF

报告写作。要求：主要思路有明确的说明，重点代码有详细的注释，行文逻辑清晰可读性强，报告整体写作较为专业。 **(20 分)**

说明：

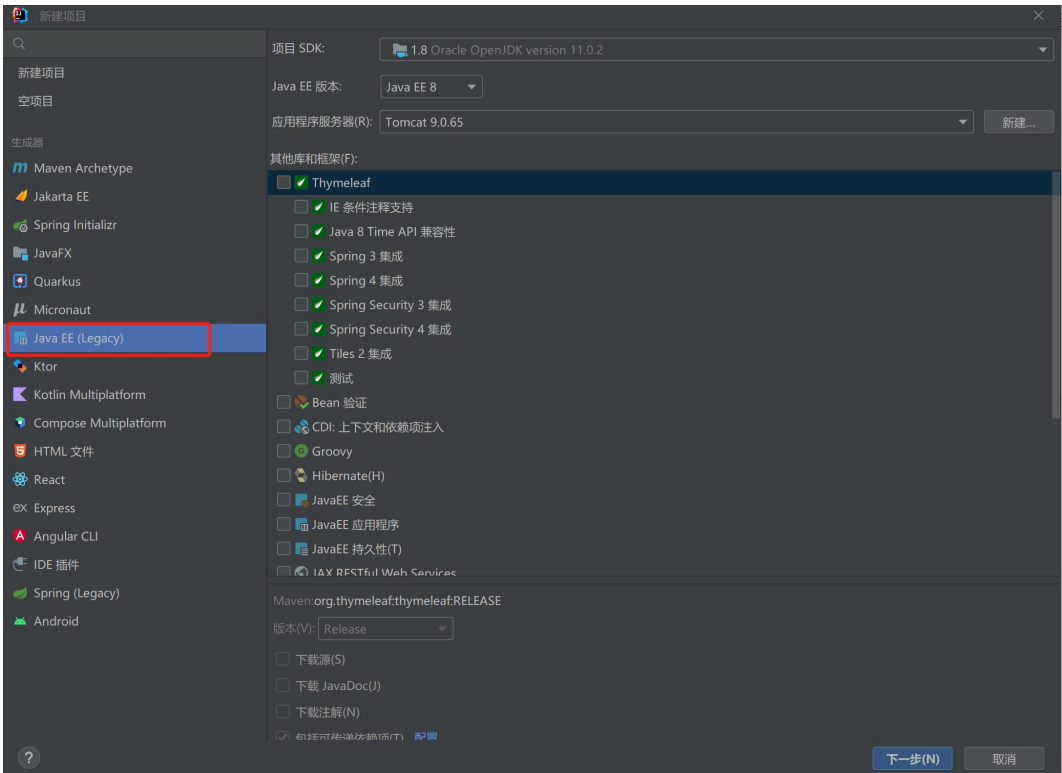
- (1) 本次实验课作业满分为 100 分，占总成绩的比例（待定）。
- (2) 报告正文：请在指定位置填写，本次实验**需要提交实验报告+源码**，其中，**源码按工程包（或文件夹）路径不变压缩成一个.rar(或.zip)文件提交，实验报告不要压缩进压缩包文件。**
- (3) 个人信息：**WORD 文件名中的“姓名”、“学号”，请改为你的姓名和学号**；实验报告的首页，请准确填写“学院”、“专业”、“报告人”、“学号”、“班级”、“实验报告提交时间”等信息。
- (4) 提交方式：截至时间前，请在 Blackboard 平台中提交。
- (5) 发现雷同，所有雷同者该次作业记零分。
- (6) 期末考试阶段补交无效。

三、实验过程及结果

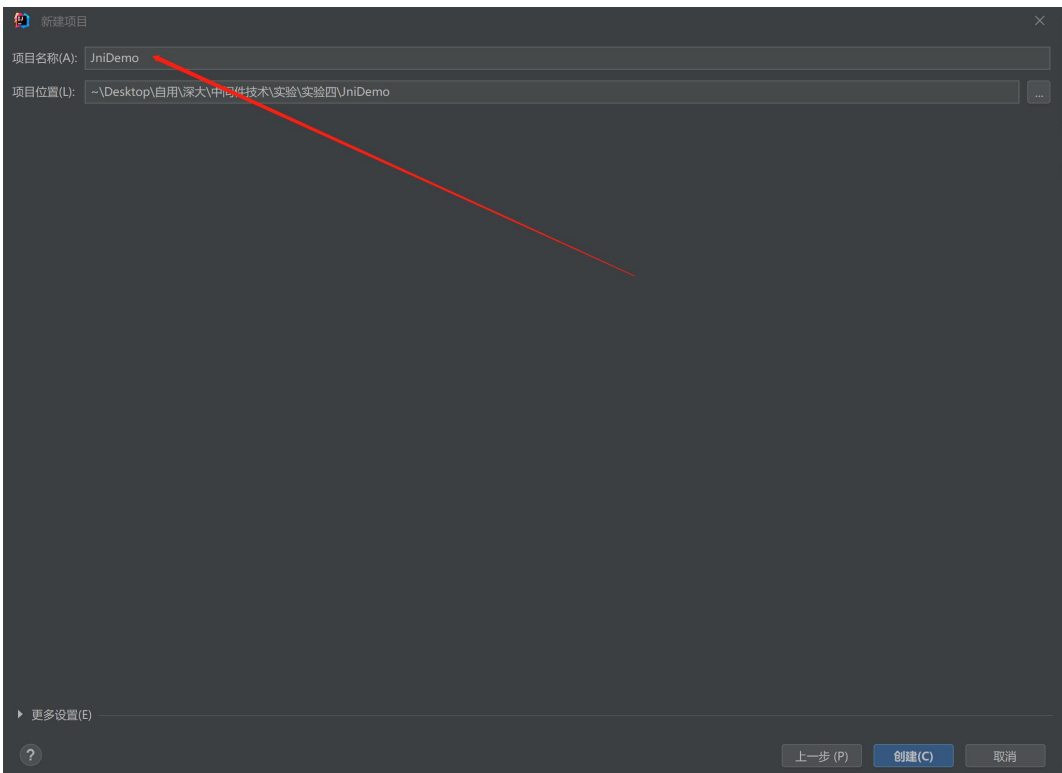
1. 创建 java 工程

1.1. 新建 java 工程

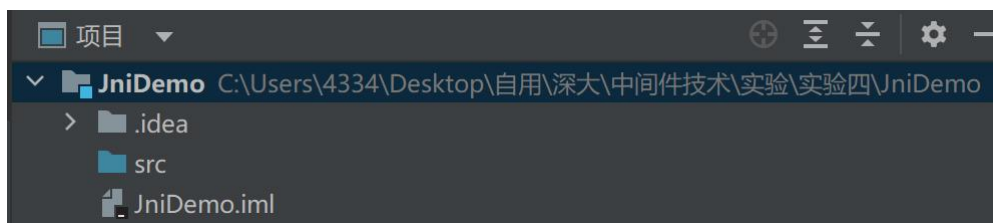
1.1.1. 在 IDEA 中选择新建项目，选择新建为 JavaEE 项目，并点击下一步。



1.1.2. 输入项目名并选择项目地址，点击创建。至此，java 工程的创建完成。

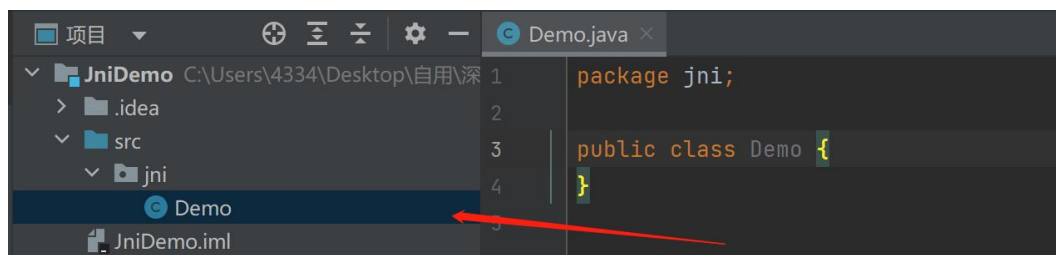


1.1.3. 完成创建项目目录如下：



1.2. 编写对应 C/C++程序的 java 对象

1.2.1. 在上述目录中的 src 文件夹下创建 jni 包，新建类 Demo.class:

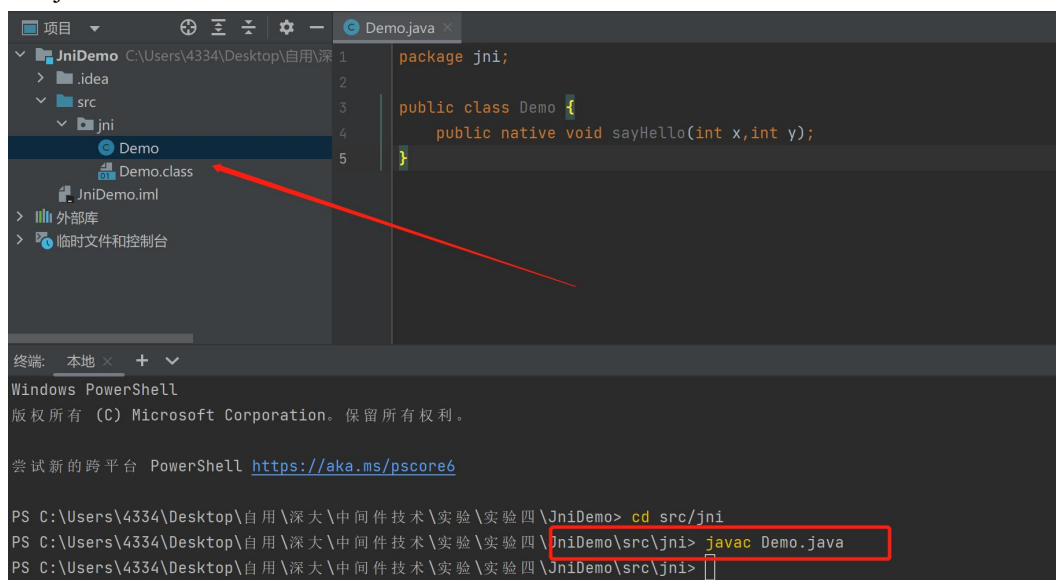


内容如下:

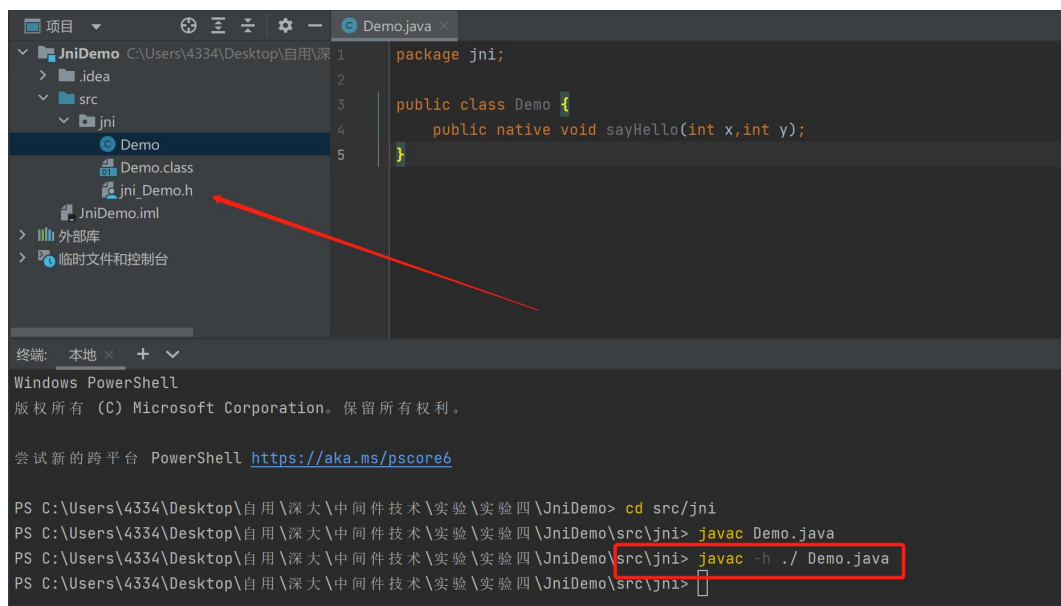
```
package jni;

public class Demo {
    public native void sayHello(int x,int y);
}
```

1.2.2. 在终端 Terminal 中生成 class 类文件: 进入 demo.java 类所在文件夹, 输入命令 “javac Demo.java”, 得到 class 文件。



1.2.3. 同样在终端中输入命令 “javac -h ./ Demo.java”, 生成 h 头文件:



h 头文件的内容如下:

```
/* DO NOT EDIT THIS FILE - it is machine generated */
#include <jni.h>
/* Header for class jni_Demo */

#ifndef _Included_jni_Demo
#define _Included_jni_Demo
#ifdef __cplusplus
extern "C" {
#endif
/*
 * Class:      jni_Demo
 * Method:     sayHello
 * Signature:  (II)V
 */
JNIEXPORT void JNICALL Java_jni_Demo_sayHello
    (JNIEnv *, jobject, jint, jint);

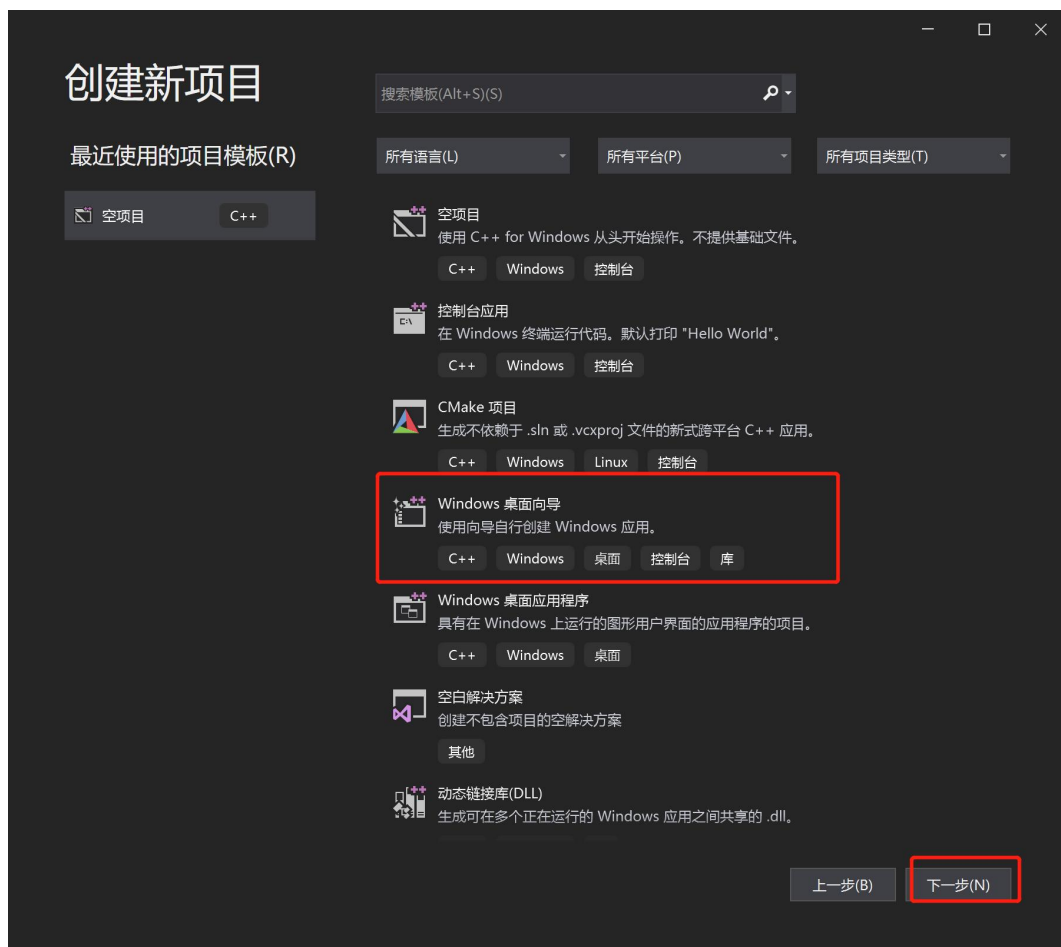
#ifdef __cplusplus
}
#endif
#endif
```

需注意: 该文件属于自动生成, 作为 C++ 文件和 java 文件的关联, 编译制作 dll 动态链接库也需要用到, 不能进行改动。
至此, java 对应的接口方面内容完成。

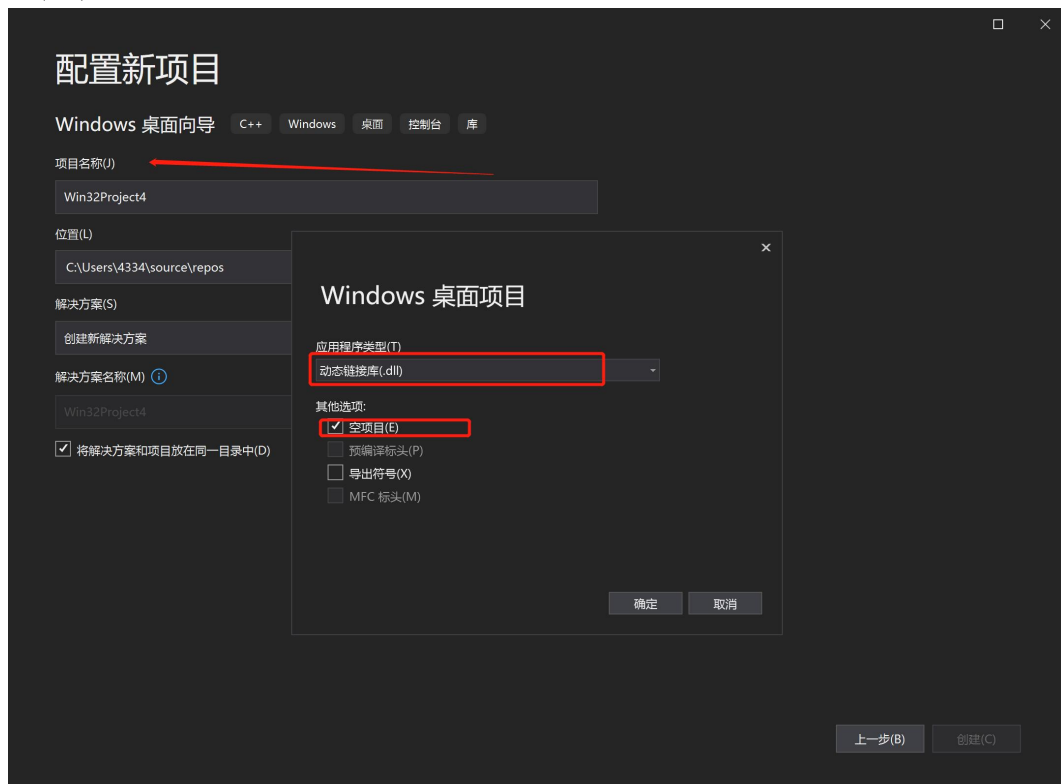
2. 编译制作 dll 动态库

2.1. 创建 dll 工程

2.1.1. 打开 VS, 选择新建项目, 选中 Windows 桌面向导, 点击下一步。



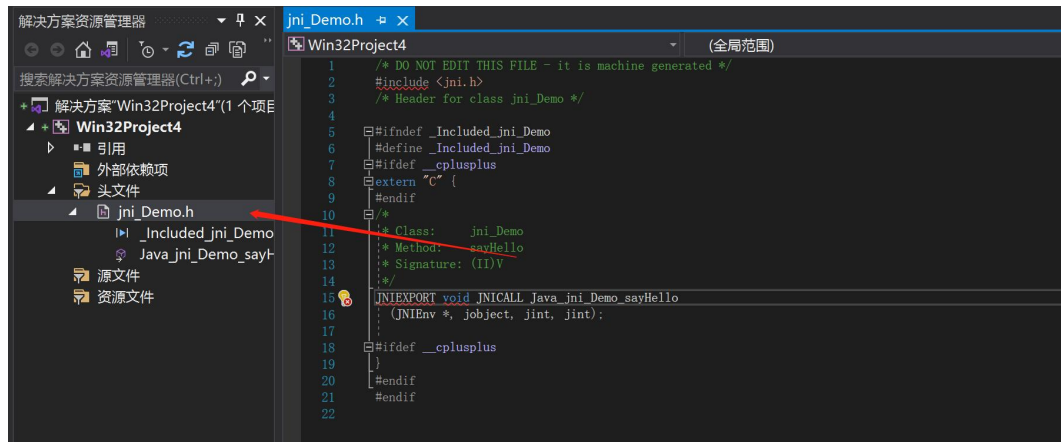
2.1.2. 输入项目名并选择项目位置，点击创建。在弹出窗口“应用程序类型”一栏选择“动态链接库(.dll)”，勾选空项目，点击确定。



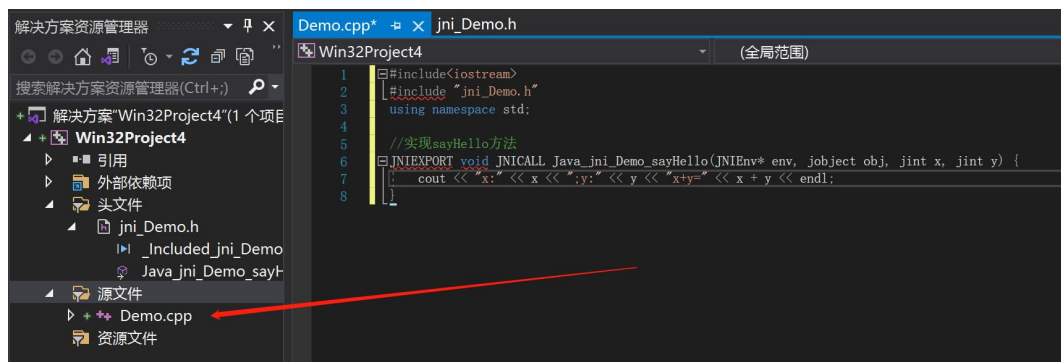
至此，dll 工程创建完成。

2.2. 编写 C/C++源文件

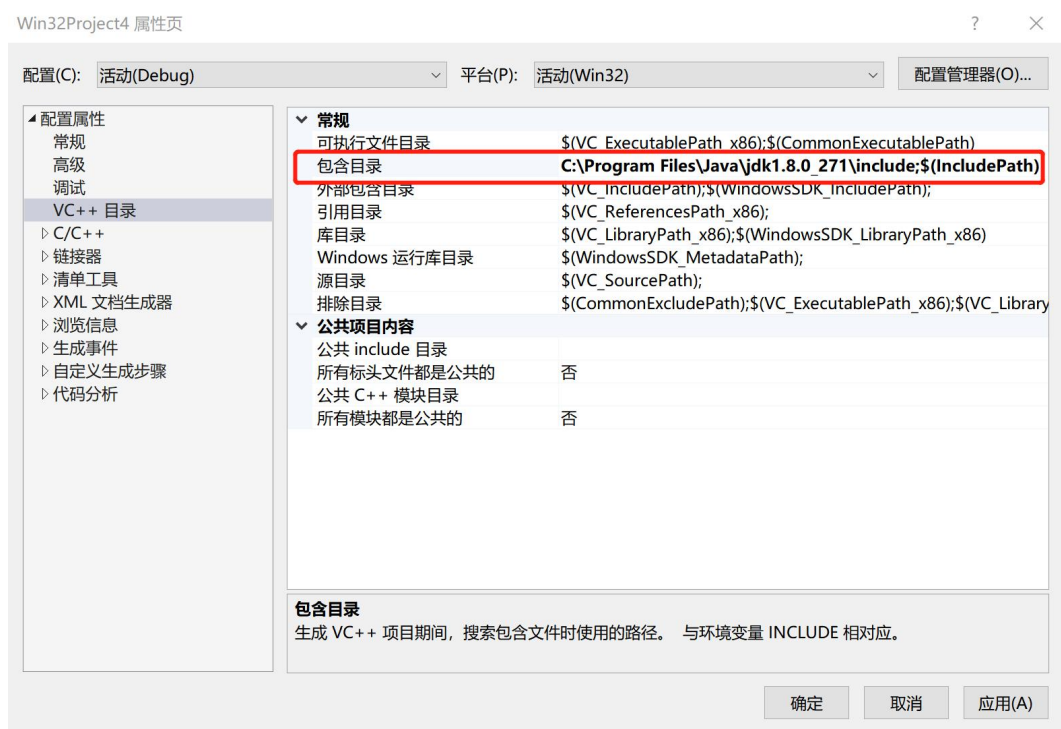
2.2.1. 将 1.2.3 中生成的 jni_Demo.h 文件复制到项目中。



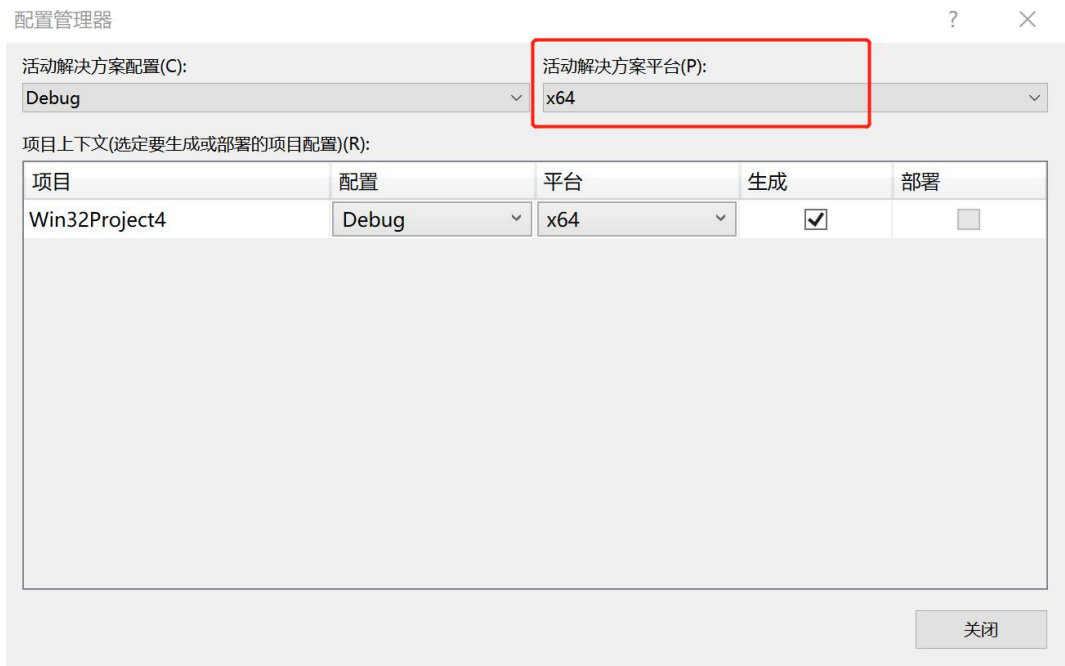
2.2.2. 新建一个 cpp 文件作为接口实现 cpp 文件，命名为“Demo.cpp”，并编写代码内容如下。



2.2.3. 配置包含目录: 右击项目，选择“属性”，在 VC++ 目录→包含目录中，添加 java JDK include 文件如下（对应电脑中 JDK 安装路径中的 include 文件路径）



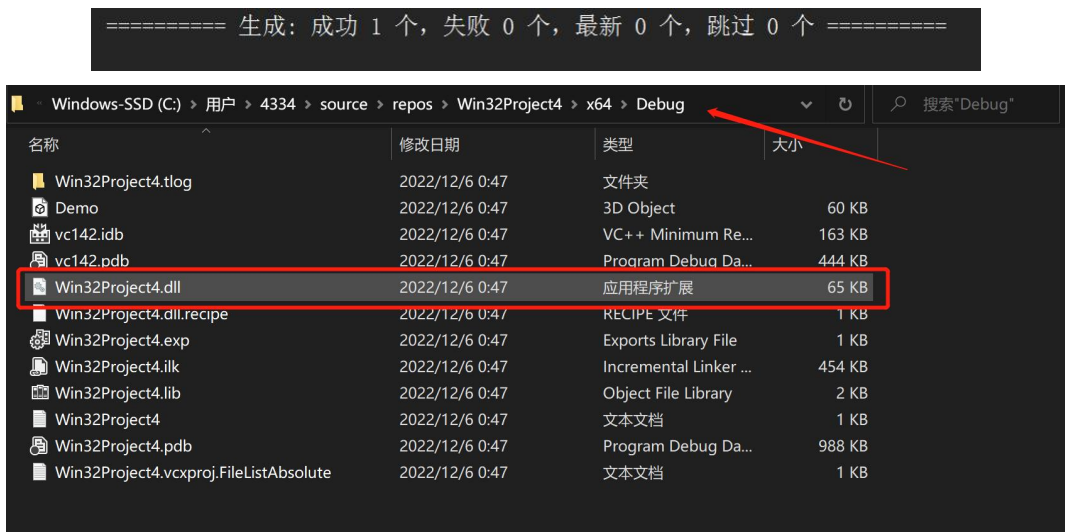
2.2.4. 配置 dll 对应的系统版本：打开配置管理器，新建解决方案平台为 X64（与系统位数对应）即可。



完成相关配置后，2.2.1 及 2.2.2 文件中的报错也能得到解决。

2.3. 生成 DLL 动态链接库文件

运行 Demo.cpp 文件，在 Debug 文件夹中生成 dll 文件如下所示。



至此，DLL 动态链接库文件生成完成。

3. Java 程序调用

3.1. 程序入口

3.1.1. 回到 IDEA 中，在 jni 文件夹中新建程序入口类 Main 类，编写调用 C++代码的代码如下。

```
package jni;  
  
public class Main {  
    public static void main(String[] args){
```



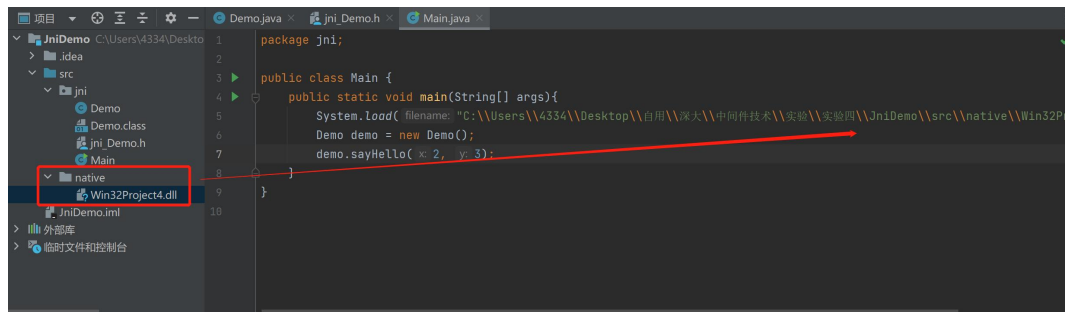
```

        System.load("C:\\Users\\4334\\Desktop\\自用\\深大\\中间件技术\\实验\\实验四\\JniDemo\\src\\native\\Win32Project4.dll");
        Demo demo = new Demo();
        demo.sayHello(2, 3);
    }

```

在这段代码中首先加载了 dll 动态链接库，通过调用 demo 对象的方法，实现 C++代码的调用。

3.1.2. 将 2.3 中生成的 dll 动态链接库文件添加到程序的某路径下，修改 load()中的文件路径为实际路径。



3.2. 运行

运行上述 Main.java 文件，可以看到输出为：

```

D:\Java\jdk\bin\java.exe "-javaagent:D:\IDEA\IntelliJ IDEA 2022.2.2\lib\idea_rt.jar=58814:D:\IDEA\IntelliJ IDEA 2022.2.2\bin" -Dfile.encoding=UTF-8 -classpath C:\Users\4334\Desktop\自用\深大\中间件技术\实验\实验四\JniDemo\out\production\JniDemo jni.Main
x:2;y:3x+y=5
进程已结束，退出代码0

```

可以看到，程序运行过程中加载了 C++的 dll 动态链接库，Java 应用程序可以正常调用 C++实现。

四、实验总结与体会

通过本次实现，学会了在面对开发项目使用混合编程的情况下，如何使用 Java JNI 解决异构语言编程问题，并在程序运行中正确调用 C++程序。也通过对该异构语言编程互操作问题的解决，对项目开发过程会遇到的问题有了更多的认识。

五、成绩评定及评语

1.指导老师批阅意见:

2.成绩评定:

指导教师签字: 毛斐巧
2022 年 月 日