[JNI编程之如何传递参数（一）——String参数的传递](http://blog.csdn.net/wangkr111/article/details/7883461)

标签： [string](http://www.csdn.net/tag/string" \t "_blank)[jni](http://www.csdn.net/tag/jni)[编程](http://www.csdn.net/tag/%e7%bc%96%e7%a8%8b)[outofmemoryerror](http://www.csdn.net/tag/outofmemoryerror)[java](http://www.csdn.net/tag/java)[input](http://www.csdn.net/tag/input)

2012-08-19 16:11 16785人阅读 [评论](http://blog.csdn.net/wangkr111/article/details/7883461#comments)(5) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/wangkr111/article/details/7883461#report)

http://static.blog.csdn.net/images/category_icon.jpg 分类：

JNI（12） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpgJava（25） http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为博主原创文章，未经博主允许不得转载。

**[java]** [view plaincopyprint?](http://blog.csdn.net/wangkr111/article/details/7883461)

1. <span style="font-weight: bold;font-size:24px; ">String参数的传递</span>

Java的String和C++的string是不能对等起来的，所以处理起来比较麻烦。先看一个例子，

**[java]** [view plaincopyprint?](http://blog.csdn.net/wangkr111/article/details/7883461)

1. <span style="font-size:14px;">**class** Prompt {
2. // native method that prints a prompt and reads a line
3. **private** **native** String getLine(String prompt);
4. **public** **static** **void** main(String args[]) {
5. Prompt p = **new** Prompt();
6. String input = p.getLine("Type a line: ");
7. System.out.println("User typed: " +input);
8. }
9. **static** {
10. System.loadLibrary("Prompt");
11. }
12. }</span>

在这个例子中，我们要实现一个native方法  
String getLine(String prompt);  
读入一个String参数，返回一个String值。

通过执行javah -jni得到的头文件是这样的

**[cpp]** [view plaincopyprint?](http://blog.csdn.net/wangkr111/article/details/7883461)

1. #include<jni.h>
2. #ifndef\_Included\_Prompt
3. #define\_Included\_Prompt
4. #ifdef \_\_cplusplus
5. **extern**"C" {
6. #endif
7. JNIEXPORT jstringJNICALL Java\_Prompt\_getLine(JNIEnv \*env, jobject **this**, jstring prompt);
8. #ifdef \_\_cplusplus
9. }
10. #endif
11. #endif

jstring是JNI中对应于String的类型，但是和基本类型不同的是，jstring不能直接当作C++的string用。如果你用  
cout << prompt << endl;  
编译器肯定会扔给你一个错误信息的。

其实要处理jstring有很多种方式，这里只讲一种我认为最简单的方式，看下面这个例子，

**[cpp]** [view plaincopyprint?](http://blog.csdn.net/wangkr111/article/details/7883461)

1. #include"Prompt.h"
2. #include<iostream>
3. JNIEXPORT jstringJNICALL Java\_Prompt\_getLine(JNIEnv \*env, jobject obj, jstring prompt)
4. {
5. **const** **char**\* str;
6. str = env->GetStringUTFChars(prompt, **false**);
7. **if**(str == NULL) {
8. **return** NULL; /\* OutOfMemoryError already thrown \*/
9. }
10. std::cout << str << std::endl;
11. env->ReleaseStringUTFChars(prompt, str);
12. **char**\* tmpstr = "return string succeeded";
13. jstring rtstr = env->NewStringUTF(tmpstr);
14. **return** rtstr;
15. }

在上面的例子中，作为参数的prompt不能直接被C++程序使用，先做了如下转换  
str = env->GetStringUTFChars(prompt, false);  
将jstring类型变成一个char\*类型。

返回的时候，要生成一个jstring类型的对象，也必须通过如下命令，  
jstring rtstr = env->NewStringUTF(tmpstr);

这里用到的GetStringUTFChars和NewStringUTF都是JNI提供的处理String类型的函数，还有其他的函数这里就不一一列举了。