



Stone_OverLooking的博客

≡ 目录视图

≡ 摘要视图

RSS 订阅

个人资料



裸睡的蚰蚰

关注

发私信



访问：17929次

积分：1009

等级：

BLOG > 4

排名：千里之外

原创：77篇

转载：48篇

图灵赠书——程序员11月书单 【思考】Python这么厉害的原因竟然是！ 感恩节赠书：《深度学习》等另
作译者评选启动！ 每周荐书：京东架构、Linux内核、Python全栈

关于android studio使用cmake 编写jni调用静态库、动态库的方式

2017-11-30 17:31

34人阅读

评论(0)

收藏

分享

≡ 分类：

C++ (29) ▼ JNI (6) ▼

版权声明：本文为博主原创文章，未经博主允许不得转载。

最近在帮助项目的大神测试编译出来的安卓动态库，需要用到jni
(懒得)，毕竟库太多了。然后以前听说android studio被谷歌搞得可以

去编译源码库，于是乎，就更新了许多未用的android studio。在这
注册函数时，总是注册不过，最后就是用android studio原声的

XDowel

首页 注册 解决方案 帮助中心 联系我们

轻松玩转流量

整合分发

流量红包

验证码短信接口

流量红包

评论： 4条

[illegible]

- android手势 (1)
- android多媒体 (12)
- 服务器 (2)
- webRTC服务器搭建 (2)
- android音频 (7)
- android视频 (4)
- android图片之Bitmap (5)
- android基础知识学习笔记 (26)
- android之Service (1)
- JNI (7)
- Java web学习笔记 (5)
- xml (4)
- javaScript (2)
- ubuntu (8)
- servlet笔记 (5)
- JSP (6)
- Linux (17)
- C++ (30)

http://blog.csdn.net/Stone_OverLooking/article/details/78678310

在CmakeList.txt文件中需要做一些配置更改。

这个是项目打开时原版本的CmakeList.txt在这个基础上加上自己所需要的静态库

```

01. <span style="font-size:12px;">cmake_minimum_required(VERSION 3.4.1)
02.
03. #将静态库的头文件添加进来
04.     include_directories(
05.         ${CMAKE_SOURCE_DIR}/src/main/include} #头文件多的话可以继续往下加
06.         ${CMAKE_SOURCE_DIR}/src/main/include/..}
07.
08.         ${CMAKE_SOURCE_DIR}/src/main/include/...}
09.
10.         . . . .
11.     )
12.     add_library( # 自己的库名字
13.         native-lib
14.
15.         # 将自己的库编译成动态库
16.         SHARED
17.
18.         # 自己的cpp源码所在的路径即jni源码的路径
19.         src/main/cpp/native-lib.cpp )
20.
21. #添加自己所需的静态库 库多的话，可以使用下面的方式添加
22.     add_library( event      #库名字 去掉了 lib 与 .a
23.         STATIC      #必须的
24.         IMPORTED    #必须的
25.     )

```

关闭



展开

| | |
|------------------------------|--------|
| android surfaceView手势放大缩小... | (1394) |
| VS 2015添加静态库.lib文件的... | (696) |
| C++文件读写之获取文件大小... | (622) |
| 基于webrtc的apprtc服务器搭建... | (620) |
| libsrt库的编译与测试方法 | (467) |
| 基于webrtc的apprtc服务器搭建... | (424) |
| 在android jni中添加输出日志Log | (356) |
| Eclipse中测试某单独的方法 | (329) |
| android studio 编写JNI时碰到的.. | (308) |
| UML的9种图例解析 | (285) |

- android surfaceView手势放大缩... (2)
- android studio编写jni成功 (1)
- 模块通知的7种技术 (1)
- 基于webrtc的apprtc服务器搭建... (0)
- Bitmap、Matrix使用 (0)

```

26. set_target_properties(
27.     event
28.     PROPERTIES IMPORTED_LOCATION
29.     ${CMAKE_SOURCE_DIR}/libs/libevent.a #库在的路径，以cmakeList.txt所在的路径起
30. )
31.
32. #添加自己所需的静态库 库多的话，可以使用下面的方式添加
33. add_library( event      #库名字 去掉了 lib 与 .a
34.             STATIC      #必须的
35.             IMPORTED    #必须的
36. )
37. set_target_properties(
38.     event
39.     PROPERTIES IMPORTED_LOCATION
40.     ${CMAKE_SOURCE_DIR}/libs/libevent.a #库在的路径，以cmakeList.txtf
41.
42. )
43.
44.
45.
46. find_library( log-lib log ) target_link_libraries( native-lib #我自己的需要生成的动态库
47.     event #将所需的库和自己要生成的动态库连接起来，多的话一次向下追加 x264 faac ${log-
48.
49.
50.
51.
52. #include <jni.h>
53. #include <string>
54. #include <stdio.h>
55.
56. #include "ares.h"
57. #include "faac.h"
58. #include "uuid.h"
59. #include "turbojpeg.h"
60. #include "x264.h"
61.
62.

```

在native-lib.cpp文件中写上的是关于自己的jni函数，将所需头文件包含进来，就可以通过jni函数调用静态库的函数。

关闭



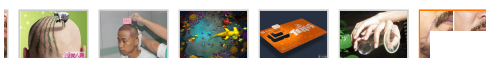
- JNI相关的函数使用方式 (0)
- 使用Nginx搭建代理服务器通过.. (0)
- 使用内置的Gallery选择图片 (0)
- 调用相机，自定义拍照 Runnable (0)
- TS流格式文件解析 (0)

推荐文章

- * 【2017年11月27日】CSDN博客更新周报
- * 【CSDN】邀请您来GitChat赚钱啦！
- * 【GitChat】精选——JavaScript进阶指南
- * 改做人工智能之前，90%的人都没能给自己定位
- * TensorFlow 人脸识别网络与对抗网络搭建
- * Vue 移动端项目生产环境优化



胡子种植



```

63. void insertSort(int *data,int len);
64. //调用libcares的例子
65. extern "C"
66. JNIEXPORT jint JNICALL Java_com_skylight_testlibs_JNIHeader_AreasLibInit(JNIEnv *env, j
67. {
68.     jint result=ares_library_init(1);
69.     return result;
70. }
71. //调用libuuid的例子
72. extern "C"
73. JNIEXPORT jint JNICALL Java_com_skylight_testlibs_JNIHeader_UuidIsNull(JNIEnv *env, j
74. {
75.     const uuid_t uu= {0};
76.     jint result=uuid_is_null(uu);
77.     return result;
78. }
79.
80. //调用libturbojpeg的例子
81. extern "C"
82. JNIEXPORT jint JNICALL Java_com_skylight_testlibs_JNIHeader_TJAlloc(JNIEnv
83. {
84.     unsigned char* c= tjAlloc(128);
85.     const char* result=(const char*)c;
86.     if (c == NULL)
87.     {
88.         return -1;
89.     } else{
90.         return 0;
91.     }
92. }
93.
94. //调用libx264库的例子
95. extern "C"
96. JNIEXPORT jint JNICALL Java_com_skylight_testlibs_
97. {
98.     x264_picture_t *ipic= nullptr;
99.     x264_picture_init( ipic );
100.    jint result= x264_picture_alloc( ipic, 32, 512,
101.    return result;

```

关闭

轻松玩转流量
整合分发

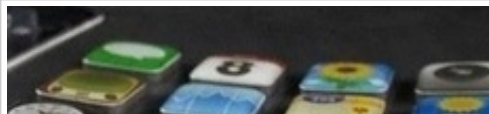
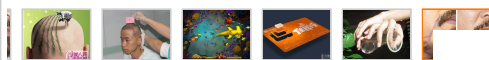
验证码短信接口



开发一个app多少钱



胡子种植



```

102.     }
103.     //调用faac库的函数例子
104.     extern "C"
105.     JNIEXPORT jint JNICALL Java_com_skylight_testlibs_JNIHeader_FaacEncClose(JNIEnv *env,
106.     {
107.         jint result=-2;
108.         unsigned long nSampleRate = 44100;
109.         unsigned int nChannels = 2;
110.         unsigned int nPCMBitSize = 16;
111.         unsigned long nInputSamples = 0;
112.         unsigned long nMaxOutputBytes = 0;
113.         faacEncHandle hEncoder = {0};
114.         faacEncHandle handle= faacEncOpen(nSampleRate, nChannels, &nInputSamp:
115.         if (handle == NULL)
116.         {
117.             result= -1;
118.         } else{
119.             result= 0;
120.         }
121.         return result;
122.     }
123.
124.     //调用insertSort函数
125.     extern "C"
126.     JNIEXPORT void JNICALL Java_com_skylight_testlibs_JNIHeader_insertSort(JNIEnv *env,
127.     {
128.         jint *data=env->GetIntArrayElements(array, NULL);
129.         jsize len=env->GetArrayLength(array);
130.         insertSort(data, len);
131.         env->ReleaseIntArrayElements(array, data, 0);
132.     }
133.     //排序
134.     void insertSort(int *data, int len)
135.     {
136.         int temp = 0;
137.         int i = 1;
138.         int j = 0;
139.         for (; i < len; i++)
140.         {

```

关闭

轻松玩转流量
整合分发

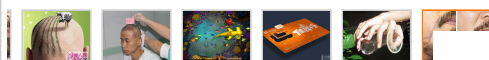
验证码短信接口



开发一个app多少钱



胡子种植



```

141.         temp = data[i];
142.         j = (i - 1);
143.         for (; j >= 0; j--)
144.         {
145.             if (data[j] > temp)
146.             {
147.                 data[j + 1] = data[j];
148.             }
149.             else
150.             {
151.                 break;
152.             }
153.         }
154.         data[j + 1] = temp;
155.     }
156. }

```

在java层的代码就很简单了直接写上native方法就行可以了。

```

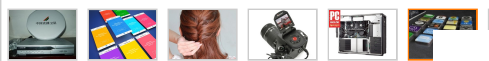
162. public class JNIHeader
163. {
164.     static {
165.         System.loadLibrary("native-lib");
166.     }
167.     public native void insertSortArray(int[] data);
168.     public native int AreasLibInit();
169.     public native int UuidIsNull();
170.     public native int TJAlloc();
171.     public native int X264PictureAlloc();
172.     public native int FaacEncClose();
173. }
174.
175. 在build.gradle中还需要加入以下配置，在defaultConfig中加入
176.
177.
178. externalNativeBuild {
179.     cmake {

```

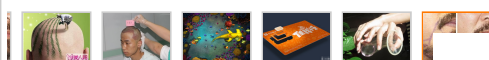
关闭



开发一个app多少钱



胡子种植



```

180.         arguments '-DANDROID_PLATFORM=android-23',
181.                   '-DANDROID_TOOLCHAIN=clang', '-DANDROID_STL=gnustl_static'
182.     }
183. }
184.
185.     ndk {
186.         abiFilters 'armeabi-v7a' // armeabi-v7a 这里需要过滤一下处理器类型
187.     }
188.

```

在defaultConfig的下面加入这个配置

```

191.     sourceSets {
192.         main {
193.             jniLibs.srcDirs = ['libs']
194.         }
195.     }
196.

```

这样就可以使用cmake在android studio中调用静态库的函数了

[cpp]

01. android studio在调用动态库时和调用静态库的方式略有不同,不知道别人是
02. 怎么解决的,目前我的android studio碰到调用动态库的问题如下

03. 1、在src/main/目录下新建一个文件夹jniLibs,在该文件夹中需要
04. -->即armeabi-v7a armeabi-arm 等子文件夹,最终的目录

05. ----->src

06. ----->main

07. ----->jniLibs

08. ----->armeabi-v7a (armeabi-arm)

09. 路径会是--->src/main/jniLibs/armeabi-v7a 或者对应的

10. 2、将自己所要调用的动态库.so文件放置在上述目录中 例如(src/main/

11. 3、在CmakeList.txt文件中添加动态库

12. add_library(库名 SHARED IMPORTED)

13.

14. set_target_properties(库名 PROPERTIES

15. IMPORTED_LOCATION \${CMAKE_S

库名.so)

关闭

轻松玩转流量
整合分发

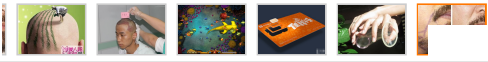
验证码短信接口



开发一个app多少钱



胡子种植



```
16.      注:ANDROID_ABI就是ABI类型--->armeabi-v7a或者对应的
17.
18.      4、在java代码中加载动态库
19.      System.loadLibrary("库名");
20.      建议:将调用的动态库放在native-lib库的上面。先加载三方动态库
21.
22.      5、在target_link_libraries( )中将动态库添加进去
```

最近在帮助项目的大神测试编译出来的安卓动态库，需要用到jni，但是使用ndk去编译有太麻烦(还是懒)，毕竟月前听说android studio被谷歌搞得可以使用cmake

去编译源码库，于是乎，就更新了许多未用的android studio。在这个过程中真的是很恶心。使用jni动态注册函数过，最后就是用android studio原声的

native-lib.cpp通过静态注册的方式搞定。

在CmakeList.txt文件中需要最一些配置更改。

这个是项目打开时原版本的CmakeList.txt在这个基础上加上自己所需要的静态库

顶 0 踩 0

- 上一篇 关于faac的一点使用方式

关闭

XDowel

首页 注册 登录 联系我们 帮助中心 联系我们

轻松玩转流量

整合分发

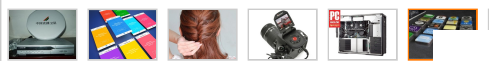
分秒流量

流量红包

验证码短信接口



开发一个app多少钱



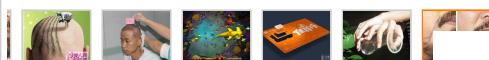
- [下一篇](#) JNI相关的函数使用方式

相关文章推荐

- VS2015 Android JNI 静态库 动态库 APK 本地调试D..
- Jni测试引用第三方静态库和动态库demo
- 腾讯云容器服务架构实现介绍--董晓杰
- 用Word2Vec处理自然语言
- windows下使用cmake和ndk编译android所用动态库...
- 使用Android JNI调用第三方静态库并且生成静态库...
- 微博热点事件背后的数据库运维心得--张冬洪
- Java之优雅编程之道
- CMake 创建静态库及动态库供函数调用
- Android Studio cmake方式jni开发，在c层...
- JDK9新特性--Array
- Jni中执行ndk-build，编译第三方静态库(.a)
- 《CMake实践》笔记三：构建静态库与动态库 及 如..
- 使用gcc建立并调用静态库与动态库
- Kubernetes容器云平台实践--李志伟
- cmake ---动态库与静态库



胡子种植



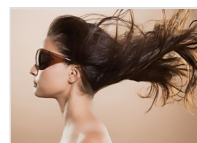
种头发原理



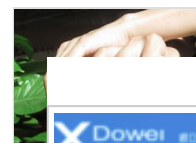
种头发危害



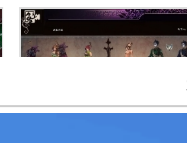
胡子种植



种植头发视频



润



关闭

查看评论

暂无评论

发表评论

用户名： weixin_35068028

轻松玩转流量
整合分发

验证码短信接口

流量红包

流量统计

流量分析

流量监控

流量预警

流量优化

流量提升

流量变现

流量运营

流量推广

流量营销

流量管理

流量服务

流量支持

流量平台

流量工具

流量资源

流量数据

流量报告

流量趋势

流量对比

流量排名

流量评价

流量反馈

流量改进

流量创新

流量突破

流量飞跃

流量爆发

流量增长

流量稳定

流量安全

流量可靠

流量透明

流量公正

流量公开

流量共享

流量共赢

流量合作

流量联盟

流量生态

流量未来

流量梦想

流量希望

流量信心

流量决心

流量恒心

流量耐心

流量信心

流量决心

流量恒心

流量耐心



评论内容：



提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

[公司简介](#) | [招贤纳士](#) | [广告服务](#) | [联系方式](#) | [版权声明](#) | [法律顾问](#) | [问题报告](#) | [合作伙伴](#) | [论坛反馈](#)[网站客服](#)[杂志客服](#)[微博客服](#)webmaster@csdn.net

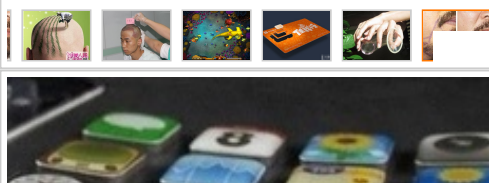
400-660-0108

| 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved



胡子种植



关闭

XDowel

首页 (当前) 解决方案 项目合作 品牌合作

联系我们

轻松玩转流量

整合分发

分网流量

流量红包

自主研发 功能强大 秒速上线 VPS定制

验证码短信接口