

原

Android NDK使用之--用pthread实现多线程

2015年12月16日 21:23:02

jike0901xuye

阅读数：3812

Android系统本身包括Java和Linux内核；因此在Android应用中实现多线程就可以是基于Java上的和基于Linux上的；今天本文介绍的就是Android利用Linux下的pthread来实现多线程。既思路如下：

我们将异步任务需要实现的方法在Java中定义，但是运行异步任务的线程交给pthread实现；

因此首先设计一个接口，作为异步任务的业务接口：

```
1 // 异步任务业务接口
2 public interface LocalRunnable {
3     public void run();
4 }
```

定义本地方法

```
1
2 /**
3  *
4  * @author 徐晔 线程池
5  */
6 public class ThreadPool {
7     static {
8         System.loadLibrary("threadpool");
9     }
10    /** 初始化本地线程池 */
11    public native void init();
12    /** 本地线程中执行任务 */
13    public native int run(LocalRunnable runnable);
14    /**释放资源*/
15    public native void recycle();
16 }
```

通过本地方法类生成的.h文件如下：

```
1 com_localThread_ThreadPool.h
2 /* DO NOT EDIT THIS FILE - it is machine generated */
3 #include <jni.h>
4 //实现pthread需要包括的接口
5 #include <unistd.h>
6 #include <pthread.h>
7 /* Header for class com_localThread_ThreadPool */
```

```

10 #define _Included_com_localThread_ThreadPool
11 #ifdef __cplusplus
12 extern "C" {
13 #endif
14 /*
15  * Class:      com_localThread_ThreadPool
16  * Method:     init
17  * Signature:  ()V
18  */
19 JNIEXPORT void JNICALL Java_com_localThread_ThreadPool_init(JNIEnv *, jobject);
20
21 /*
22  * Class:      com_localThread_ThreadPool
23  * Method:     run
24  * Signature:  (Lcom/localThread/LocalRunnable;)I
25  */
26 JNIEXPORT jint JNICALL Java_com_localThread_ThreadPool_run(JNIEnv *, jobject,
27     jobject);
28
29 /*
30  * Class:      com_localThread_ThreadPool
31  * Method:     recycle
32  * Signature:  ()V
33  */
34 JNIEXPORT void JNICALL Java_com_localThread_ThreadPool_recycle(JNIEnv *,
35     jobject);
36
37 #ifdef __cplusplus
38 }
39 #endif
40 #endif

```

具体的实现代码如下：

```

1  /* DO NOT EDIT THIS FILE - it is machine generated */
2  #include "com_localThread_ThreadPool.h"
3  /* Header for class com_localThread_ThreadPool */
4
5  static JavaVM *jvm = NULL;
6  static jobject jobj = NULL;
7  static jmethodID mid = NULL;
8  static int flag = -1;
9  static pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
10 static pthread_cond_t cond = PTHREAD_COND_INITIALIZER;
11 // 在加载库时执行
12 jint JNI_OnLoad(JavaVM *vm, void *reserved) {
13     jvm = vm;

```

[2019人工智能薪资](#)
[Python资料免费领](#)
[会员任意学](#)
[Java薪资多少](#)
[怎样才能不被裁员](#)
[c++游戏编程](#)
[商城app开发](#)
[登录](#)
[注册](#)
[×](#)

```
16 // 在卸载库时执行
17 void JNI_OnUnload(JavaVM *vm, void *reserved) {
18     jvm = NULL;
19 }
20 //pthread中执行的函数
21 void *nativeWork(void *args) {
22     JNIEnv *env = NULL;
23     // 将本地线程连接到虚拟机上, 这样本地线程才对虚拟机可见并且可以访问到虚拟机所在进程的资源
24     if (0 == jvm->AttachCurrentThread(&env, NULL)) {
25         while (flag == 0) {
26             if (jobj == NULL) {
27                 //进入等待
28                 pthread_cond_wait(&cond, &mutex);
29             } else {
30                 //执行方法
31                 env->CallVoidMethod(jobj, mid);
32                 env->DeleteGlobalRef(jobj);
33                 jobj = NULL;
34             }
35         }
36         jvm->DetachCurrentThread();
37     }
38     return (void *) 1;
39 }
40
41 /*
42  * Class:      com_localThread_ThreadPool
43  * Method:     init
44  * Signature:  ()V
45  */
46 JNIEXPORT void JNICALL Java_com_localThread_ThreadPool_init(JNIEnv * env,
47     jobject obj) {
48     if (NULL == mid) {
49         //缓存方法
50         jclass cls = env->FindClass("com/localThread/LocalRunnable");
51         mid = env->GetMethodID(cls, "run", "()V");
52         if (mid == NULL) {
53             return;
54         }
55         flag = 0;
56         pthread_t thread;
57         //创建新线程
58         pthread_create(&thread, NULL, nativeWork, NULL);
59     }
60 }
61
62 /*
63  * Class:      com_localThread_ThreadPool
```

[2019人工智能薪资](#)[Python资料免费领](#)[会员任意学](#)[Java薪资多少](#)[怎样才能不被裁员](#)[c++游戏编程](#)[商城app开发](#)[登录](#)[注册](#)[×](#)

```
66  */
67  JNIEXPORT jint JNICALL Java_com_localThread_ThreadPool_run(JNIEnv * env,
68      jobject obj, jobject callback) {
69      // 设置参数, 唤醒等待的线程
70      flag=0;
71      jobj = env->NewGlobalRef(callback);
72      return pthread_cond_signal(&cond);
73  }
74
75  /*
76  * Class:      com_localThread_ThreadPool
77  * Method:     recycle
78  * Signature:  ()V
79  */
80  JNIEXPORT void JNICALL Java_com_localThread_ThreadPool_recycle(JNIEnv * env,
81      jobject obj) {
82      // 释放资源
83      flag = -1;
84      env->DeleteGlobalRef(jobj);
85  }
```

Java层对本地线程类的一个封装：

```
1
2  public class ThreadPoolRun {
3
4      /** 利用阻塞队列保存任务 */
5      private static LinkedBlockingDeque<BaseLocalRunnable> runnables = new LinkedBlockingDeque<BaseLocalRunnable>();
6      private BaseLocalRunnable lr = null;
7      public static ThreadPoolRun instance = new ThreadPoolRun();
8      private ThreadPool pool = null;
9      boolean runflag = false;
10
11     private ThreadPoolRun() {
12         pool = new ThreadPool();
13         pool.init();
14     }
15
16     /** 添加任务 */
17     public void addTask(BaseLocalRunnable runnable) {
18         runnables.addLast(runnable);
19     }
20
21     /** 取出任务 */
22     private BaseLocalRunnable getTask() {
23         try {
24             return runnables.removeFirst();
```

```

27         return null;
28     }
29 }
30
31 public void runTask() {
32     synchronized (pool) {
33         lr = getTask();
34         if (lr != null && !runflag) {
35             runflag = true;
36             lr.before(lr.taskid);
37             pool.run(lr);
38         }
39     }
40
41 }
42 //当每次执行完成上一个任务后，设置标示，运行下一个任务
43 public void runend() {
44     runflag = false;
45     runTask();
46 }
47 }
--

```



实现基本任务类：

```

1
2 public abstract class BaseLocalRunnable implements LocalRunnable, MainRunnable {
3
4     private Handler mainHandler;
5     private MainRunnable mainRunnable;
6     int taskid;
7
8     public BaseLocalRunnable(Handler mainHandler, MainRunnable mainRunnable) {
9         this.mainHandler = mainHandler;
10        this.mainRunnable = mainRunnable;
11    }
12
13    @Override
14    public final void run() {
15        onfinished(dobackground(), taskid);
16    }
17
18    public final void execute(int taskid) {
19        ThreadPoolRun.instance.addTask(this);
20        this.taskid = taskid;
21        if (!ThreadPoolRun.instance.runflag) {
22            ThreadPoolRun.instance.runTask();
23        }
24    }
25 }

```

```
26
27     public abstract Object doInBackground();
28
29     @Override
30     public final void onfinished(final Object object, final int raskid) {
31         mainHandler.post(new Runnable() {
32             @Override
33             public void run() {
34                 if (mainRunnable != null) {
35                     mainRunnable.onfinished(object, raskid);
36                 }
37                 // 读取下一个信息
38                 ThreadPoolRun.instance.runend();
39             }
40         });
41
42     }
43
44     @Override
45     public final void before(int id) {
46         if (mainRunnable != null) {
47             mainRunnable.before(id);
48         }
49     }
50
51 }
```

主类需要实现的接口：

```
1
2     public interface MainRunnable {
3
4         public void before(int id);
5
6         public void onfinished(Object object, int raskid);
7
8     }
```

测试代码：

```
1
2     public class Task extends BaseLocalRunnable {
3
4         public Task(Handler mainHandler, MainRunnable mainRunnable) {
5             super(mainHandler, mainRunnable);
6         }
7     }
```

```
10     try {
11         Thread.sleep(3000);
12     } catch (Exception e) {
13         e.printStackTrace();
14     }
15     return "执行完成";
16 }
17
18 }
19
20
21 public class TestActivity extends Activity implements MainRunnable {
22
23     private TextView text;
24     private Button am_button;
25     private Handler mainHandler = null;
26     private Task task;
27
28     @Override
29     protected void onCreate(Bundle savedInstanceState) {
30         super.onCreate(savedInstanceState);
31         setContentView(R.layout.activity_main);
32         mainHandler = new Handler(getMainLooper());
33         text = (TextView) findViewById(R.id.text);
34         am_button = (Button) findViewById(R.id.am_button);
35         am_button.setOnClickListener(new OnClickListener() {
36             @Override
37             public void onClick(View v) {
38                 task = new Task(mainHandler, TestActivity.this);
39                 task.execute(0);
40             }
41         });
42
43     }
44
45     @Override
46     public void before(int id) {
47         Log.v("任务", id + "开启");
48     }
49
50     @Override
51     public void onfinished(Object object, int taskid) {
52         Log.v("结果", "结束" + object.toString());
53     }
54 }
```



运行结果如下：我连续点击了几次按钮，出现的结果是一次一次执行

L...	Time	PID	TID	Application	Tag	Text
V	12-16 21:19:44.481	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:19:47.481	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:19:47.481	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:19:50.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:19:50.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:19:53.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:19:53.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:19:56.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:19:56.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:19:59.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:19:59.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:20:02.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:20:02.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:20:05.491	18235	18235	com.xy.studyndk	结果	结束执行完成
V	12-16 21:20:05.491	18235	18235	com.xy.studyndk	任务	0开启
V	12-16 21:20:08.491	18235	18235	com.xy.studyndk	结果	结束执行完成

Java培训需要多少钱？

零基础学IT选Java，易学、高薪、前景广，100万人才缺口，互联网必备人才



想对作者说点什么

Android NDK线程篇（四）

本篇重点介绍原生线程的相关知识，包括线程基础知识，线程同步等，学完本篇，你将会对NDK原生线程有一个更...

👁 1515

来自：[kgdwbb的专栏](#)

Using pthread in Android

【原创文章，转载请保留或注明出处：http://blog.sina.com.cn/s/blog_675eba3901010tnk.html】 I have a project a...

👁 6697

来自：[yinyhy的专栏](#)

linux和android端的pthread学习

pthread lock和trylock示例代码，在linux上完成代码编写，然后适配到android设备上而不做代码的修改。...

👁 3733

来自：[dizuo的专栏](#)

2019人工智能薪资

Python资料免费领

会员任意学

Java薪资多少

怎样才能不被裁员

c++游戏编程

商城app开发

登录

注册

×

超系统的Java学习路线规划，你一定不知道的薪资翻倍秘诀！

百万程序员点赞的学习路线，你看过吗？

android下pthread相关概念理解

795

1.结构体理解 pthread_attr_t 2.create //创建线程， 成功 返回0， 出错返回-1 //第一个参数为线程id指针， ...

来自： u010029439的博客

android移植第三方库，找不到pthread

06-30

第三方库中有pthread的一些接口，但是好像android系统的源码中，没有提供POSIX的库。 结果在编译的时候报错：ld :cannot find -lpthread. 大家是怎...

c++ for Android ----- ndk 编译链接 pthread 报错

9770

好端端的c++编译/bin/ld: error: cannot find -lpthread

用ndk编写Android环境下的pthread多线程程序

6360

转载自我的新浪博客 http://blog.sina.com.cn/s/blog_675eba3901010tnk.html

来自： 锐之锋芒

android bionic缺失pthread_cancel的解决方法

4805

在native code中使用多线程好处多多，但是Android的bionic并没有完全实现标准POSIX线程库的所有API，例如pthr...

来自： yzx41099296的专栏

用它兑水喝！六旬老汉都能坚持一小时不射！何况你！

远中宏建·熨熨

Android NDK pthreads详细使用

1347

这个pthread.h文件可以在NDK环境里创建子线程，并对线程能够做出互斥所、等待、销毁等控制。写这个博客的原...

来自： z979451341的博客

文章热词 Android Android培训 Android课程 Android学习 Android视频教程

相关热词

pthread多线程c++ c++11 多线程与pthread android ndk 版本 android ndk r13 c++多线程用互斥锁 python3菜鸟教程多线程 多线程python菜鸟教程

Android NDK and pthread

288

http://stackoverflow.com/questions/30801752/android-ndk-and-pthread http://mobilepearls.com/labs/...

来自： 张同光 (Tongguang Zhan...



pan_jinquan
207篇文章
排名:2000+

关注



菜鸟-翡翠
218篇文章
排名:829

关注



panda1234lee
441篇文章
排名:2000+

关注



miago
12篇文章
排名:千里之外

关注

android线程间通信的四种实现方式

4500

1，通过Handler机制. private void one() { handler=new Handler(){ @Override ...

来自： liuxingyuaxian的博客

2019人工智能薪资

Python资料免费领

会员任意学

Java薪资多少

怎样才能不被裁员

c++游戏编程

商城app开发

登录

注册

×

<div>Android NDK中数据类型传递使用</div> <div>根据上一篇对应的数据类型关系,从这一篇开始一种一种的对应进行测试.工程的话还是用第一篇中的那个工程: 先介...</div>		622	来自: 刘志保的博客
<div>android不同线程之间数据交互</div> <div>跨线程(不是跨进程, 一个程序间主要是跨线程, 不同程序间跨进程, 当然也跨线程) 的数据交互, 我感觉这个问题...</div>		2115	来自: d_o_n_g2的博客
<div>用它兑水喝! 六旬老汉都能坚持一小时不射! 何况你!</div> <div>新创嘉意 · 熯焱</div>			
<div>NDK使用多线程技术加速图像处理</div> <div>最近工作挺有意思的, 对优化技术进行了各种的尝试和研究, 经过测试, 多线程对图像处理优化有一定的功效, 通...</div>		3205	来自: sonikk的专栏
<div>C++也来操作线程之JNI线程创建</div> <div>Jni中使用线程虽然没有Java那么简单, 但是有些情况还是需要使用的, 当然有</div>		2727	来自: Ataraxia的专栏
<div>Android NDK 链接posix线程库</div> <div>posix pthread通常链接pthread在CMake脚本中都是这样做的 target_link_libraries(xxx pthread)然而在Android NDK...</div>		205	来自: sliver.chen的博客
<div>NDK使用技巧、多线程调用注意、ndk中的工具使用</div> <div>//NDK 使用技巧和多线程调用注意 //http://www.ibm.com/search/csass/search/?q=ndk&sn=dw&lang=zh&cc=CN&e...</div>		4808	来自: 云守护的专栏
<div><div>下载</div>android NDK实现的Hanlder-消息队列-native多线程实例</div> <div>研究一个native实现的虚拟机, 在需要同时跑多个的情景下研究多线程, 消息队列, 本实例实现了一个 消息队列 和 Handler 接口仿照 Java 层 Handler 使用起来很方便, 很强大!</div>		06-13	
<div>老中医说: 饭后用一物, 体重不过百! 北京人必看!</div> <div>楠洲 · 熯焱</div>			
<div>Android ndk 线程回调java层方法</div> <div>项目上刚好要在ndk层使用线程</div>		4758	来自: OK, let's go
<div><div>下载</div>android ndk pthread</div> <div>android ndk pthread</div>		09-28	
<div>Pthread多线程同步的几种方式</div> <div>本文来自博文: http://www.cnblogs.com/mydomain/archive/2011/08/14/2138455.html 线程的最大特点是资源的共...</div>		4873	来自: 磊啊磊
<div>OpenCV使用pthread实现多线程加速处理图像 (C++)</div> <div>OpenCV使用pthread实现多线程加速处理图像 【尊重原创, 转载请注明出处】https://blog.csdn.net/guyuealian/arti...</div>		644	来自: pan_jinquan的博客
<div><div>2019人工智能薪资</div><div>Python资料免费领</div><div>会员任意学</div><div>Java薪资多少</div><div>怎样才能不被裁员</div><div>c++游戏编程</div><div>商城app开发</div><div>登录</div><div>注册</div><div>×</div></div>			

<div>千万不要再乱喝蜂蜜了，美女亲赴深山，揭露背后惊人黑幕！</div> <div>聚优 · 顶新</div>		
<div>pthread_cancel() NDK不支持</div> <div>pthread_cancel() 我用cmake在ndk环境下编译，pthread库没有这个函数，我原以为是cmake的问题，后来查明是 N...</div>	<div>2887</div> <div>来自： 你在他乡还好吗？</div>	
<div>ndk pthread_create</div> <div>大家好！ 请问android ndk pthread_create 与main 函数 pthread_create什么区别？ 在开发过程中遇到类似这样...</div>	<div>156</div> <div>来自： shuchuang4537的博客</div>	
<div>android ndk not support pthread_cancel</div> <div>转载请注明： http://blog.csdn.net/typename/article/details/7794958 下面是google关于这个问题的说法： Try chan...</div>	<div>6841</div> <div>来自： typename 记录点滴</div>	
<div>android ndk中创建线程的注意事项</div> <div>在android ndk中创建线程是可以的，但必须注意，否则会失败，</div>	<div>1761</div> <div>来自： 晓阳 的专栏</div>	
<div>Android中Handler从子线程发消息到主线程回调消息全过程</div> <div>1.handler是干什么的？ 我们在做一些耗时操作的时候，比如网络请求或者操作数据库等等，一般都会在子线程中进...</div>	<div>965</div> <div>来自： qq_16802313的博客</div>	
<div>千万不要再乱喝蜂蜜了!美女亲赴深山，发现惊人真相！</div> <div>聚优 · 顶新</div>		
<div>Android 中 C++ Thread线程用法</div> <div>Android对Linux线程提供了C++封装Thread类，它是线程的基类。使用Thread类，需创建一个新类继承于Thread类...</div>	<div>4486</div> <div>来自： unbroken</div>	
<div>Android C++中线程同步实现</div> <div>前言Android Framework中的线程同步主要就是对pthread的mutex和condition的封装。所以学习之前，建议先了解...</div>	<div>5215</div> <div>来自： 小一的专栏</div>	
<div>Android JNI学习入门之使用C++链表结构存储数据</div> <div>学习NDK开发，一方面是掌握Java和C/C++互相调用，另一方面是掌握C/C++语言本身的基础；因此我觉得入门的...</div>	<div>251</div> <div>来自： jike0901xuye的专栏</div>	
<div>编译参数中-pthread以及-lpthread的区别</div> <div>一般情况下，我们在链接一个（文件名为libxxx.so或libxxx.a等的）库时，会使用-lxxx的方式；在Linux中要用到多线...</div>	<div>1101</div> <div>来自： origin_lee的专栏</div>	
<div>android端NDK封装DES加密</div> <div>DES定义（百度百科）DES对称加密，对称加密，是一种比较传统的加密方式，其加密运算、解密运算使用的是同...</div>	<div>584</div> <div>来自： 航行在蓝天的蚂蚱</div>	
<div>老中医说：饭后用一物，体重不过百！北京人必看！</div> <div>楠洲 · 熺燚</div>		

<div><div>用Pthread对共享内存系统进行多线程并行编程</div><div>在并行程序设计中，有针对分布式内存系统的和针对共享内存系统的。对于分布式内存系统，每个核访问其独立直...</div></div>	<div><div>207</div><div>来自：ProHanziiee</div></div>
<div><div>使用pthread实现多线程CPU+GPU计算</div><div>本次测试目的仅仅为了尝试能否使用pthread实现多线程对CPU和GPU分别进行控制：文件结构如下：ma...</div></div>	<div><div>1939</div><div>来自：Allyli0022的专栏</div></div>
<div><div>Android NDK C++中的多线程</div><div>Android NDK C++中的多线程 前言 我们都知道Android是基于Linux内核的，而Linux是遵循POSIX线程标准的，PO...</div></div>	<div><div>237</div><div>来自：yangyasong的博客</div></div>
<div><div>ndk开发之多线程同步与通信</div><div>在进行android的ndk开发时，耗时任务会用到native子线程。在pthread头文件中定义有创建子线程、互斥锁、条件...</div></div>	<div><div>671</div><div>来自：徐福记456</div></div>
<div></div>	
<div><div>Android NDK（八）：JNI多线程</div><div>今天来唠唠JNI中关于多线程。多线程，即多个线程同时工作，多线程的问题比较复杂，在实际情况下可能会面临众...</div></div>	<div><div>1878</div><div>来自：Songlcy</div></div>
<div><div>JNI线程</div><div>JNI多个本地线程进入Java层,Java层线程进入C本地函数，线程安全</div></div>	<div><div>5039</div><div>来自：shaohuazuo的专栏</div></div>
<div><div>Android线程的正确使用姿势</div><div>进程优先级（Process Priority）进程的优先级从高到低依次分为五类，越往下，在内存紧张的时候越有可能被系统...</div></div>	<div><div>2452</div><div>来自：weifeng的博客</div></div>
<div><div>关于加入NDK编译环境的方法以及如何加入PIE模块的编译选项</div><div>之前在编译一个公司的MTK平台的Android开机加密程序的时候死活编译不通过，后面发现，有一样东西没有设置：...</div></div>	<div><div>729</div><div>来自：Bruce.yang的嵌入式之旅</div></div>
<div><div>Thread Pool 线程池 使用 线程优化</div><div>在写程序时有些异步程序只执行一遍就不需要了，为了方便经常会写下面的代码 new Thread(new Runnable() { @O...</div></div>	<div><div>1062</div><div>来自：风一样的男人</div></div>
<div></div>	
<div><div>Android开发之线程池使用总结</div><div>线程池算是Android开发中非常常用的一个东西了，只要涉及到线程的地方，大多数情况下都会涉及到线程池。Andr...</div></div>	<div><div>3.7万</div><div>来自：江南一点雨的专栏</div></div>
<div><div>pthread和信号量的简单例子</div><div>#include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; #include &lt;pthread.h&gt;g...</div></div>	<div><div>73</div><div>来自：qq_34179431的博客</div></div>

<div><div>ndk_stack打印native层崩溃堆栈</div><div>ndk-stack定位native层崩溃堆栈</div></div>			<div><div><div>787</div></div></div> <div>来自：程序员张三的笔记</div>		
<div><div>在VS2010下，用C语言编写pthread多线程程序的配置</div><div>在VS2010下，用C语言编写pthread多线程程序的配置 1、从http://sourceware.org/pthreads-win32/下载pthread 的...</div></div>			<div><div><div>2192</div></div></div> <div>来自：BAOXF专栏</div>		
<div><div>pthread + socket 实现多线程多客户端群聊服务器（简易）</div><div>原创作品，请勿转载。 /***** *pthread+socket 实现多线程 多客户端群聊...</div></div>			<div><div><div>896</div></div></div> <div>来自：邱于涵的博客</div>		
<div><div><div></div><div>Android App开发入门：使用Android Studio 2.X开发环境</div><div>本书从初学者的角度出发，通过通俗易懂的语言、丰富的实例，详细介绍使用AndroidStudio2.X开发环...</div></div></div>			<div><div><div></div></div></div>		
<div><div>很黄很暴力的十个网站</div><div>13岁的北京学生张某，在去年12月27日19时新闻联播一则关于净化网络视听的新闻里，接受采访时说的话激起了轩...</div></div>			<div><div><div>74924</div></div></div> <div>来自：Kinb_huangwei的专栏</div>		
<div><div>史上最简单的 SpringCloud 教程 终章</div><div>转载请标明出处：http://blog.csdn.net/forezp/article/details/70148833 本文出自方志朋的博客 错过了这一篇，你可...</div></div>			<div><div><div>1249397</div></div></div> <div>来自：方志朋的专栏</div>		
<div><div>webstorm 2018 激活破解方法大全</div><div>webstorm 作为最近最火的前端开发工具,也确实对得起那个价格,但是秉着勤俭节约的传统美德,我们肯定是能省则省...</div></div>			<div><div><div>692660</div></div></div> <div>来自：唐大帅的编程之路</div>		
<div><div>初次简单逆向破解安卓APP</div><div>最近下载了一个直播app，因为有会员时间限制,只能只用十分钟，但最近接触了一些逆向知识，抱着尝试的态度尝...</div></div>			<div><div><div>5748</div></div></div> <div>来自：无聊看风景的博客</div>		
<div><div>Kotlin实战(一)</div><div>Kotlin实战(一) 1、简介 Kotlin 是一个用于现代多平台应用的静态编程语言，由 JetBrains 开发。Kotlin可以编译成Ja...</div></div>			<div><div><div>2470</div></div></div> <div>来自：zping0808的博客</div>		
<div><div>Webstorm 最新激活码 多种破解方式(持续更新...)</div><div>方法：License server 注册 安装完成，打开Webstorm，在弹出的License Activation窗口中选择“License server”，...</div></div>			<div><div><div>87382</div></div></div> <div>来自：老妖儿的博客</div>		
<div><div>力压 Java 与 C 的 Python 现状如何了？</div><div>转眼间，2018 年已进入倒计时阶段，回忆起这一年中编程语言界的发展，过得风生水起的语言非 Python 莫属。而...</div></div>			<div><div><div>4749</div></div></div> <div>来自：CSDN资讯</div>		
<div><div>2018最新Web前端经典面试题及答案</div><div>本篇收录了一些面试中经常会遇到的经典面试题以及自己面试过程中遇到的一些问题，并且都给出了我在网上收集...</div></div>			<div><div><div>363361</div></div></div> <div>来自：wdlhao的博客</div>		

<div>最新迅雷“应版权方要求，文件无法下载”的解决办法</div> <div>迅雷下载有的电影电视剧的时候会出现：应版权方要求，文件无法下载，或者显示迅雷任务包含违规内容 无法继续...</div>	<div>👁 231127</div> <div>来自： 徐奕的专栏</div>
<div>pyCharm最新2018激活码</div> <div>本教程对jetbrains全系列可用例：IDEA、WebStorm、phpstorm、clion等 因公司的需求，需要做一个爬取最近上映...</div>	<div>👁 1270323</div> <div>来自： 昌昌</div>
<div>webgl第26课-在图形上贴图片</div> <div>需要电子档书籍可以Q群：828202939 希望可以和大家一起学习、一起进步！！上一节课我们学习了 颜色与纹理...</div>	<div>👁 1911</div> <div>来自： 谷子的博客</div>
<div>整理了10个干净、好用的BT、磁力链搜索网站给大家</div> <div>现在越来越流行在线看视频了，但是对于我得收藏癖爱好者，还是希望可以有比较好的资源网站的，尤其是种子、...</div>	<div>👁 107685</div> <div>来自： YXAPP的技术分享</div>
<div>颈椎病的症状,Pyhton结合治疗仪让治疗与众不同</div> <div>第1部分 Python内建对象 第1章 Python对象初探 1.1 Python内的对象 1.1.1 对象机制的基石——PyObject 1.1.2 定...</div>	<div>👁 7994</div> <div>来自： john_dung的博客</div>
<div>webgl第30课-用键盘来改变视点</div> <div>需要电子档书籍可以Q群：828202939 希望可以和大家一起学习、一起进步！！所有的课程源代码在我上传的资...</div>	<div>👁 1910</div> <div>来自： 谷子的博客</div>
<div>webstorm 2018破解版本【mac适用】下载（解压直接可用）</div> <div>webstorm激活码据内部人员透露封查太严，是故分享一套三步即可破解方法：【权威】官网最新版本破解永久有效...</div>	<div>👁 10109</div> <div>来自： 赏樱看雪撸代码</div>
<div>微服务Springcloud超详细教程+实战（十）</div> <div>本人正在找深圳Java实习工作，求大佬带飞 QQ：1172796094 如在文档中遇到什么问题请联系作者 —————...</div>	<div>👁 14066</div>
<div>微服务Springcloud超详细教程+实战（八）</div> <div>如在文档中遇到什么问题请联系作者 QQ：1172796094 本人正在找深圳Java实习工作，求大佬带飞 —————...</div>	<div>👁 17274</div>
<div>史上最全Java面试题（带全部答案）</div> <div>今天要谈的主题是关于求职，求职是在每个技术人员的生涯中都要经历多次。对于我们大部分人而言，在进入自己...</div>	<div>👁 126904</div> <div>来自： 林老师带你学编程</div>
<div>【Android学习】第三章·存储容量的获取&xml格式文本的创建与解析</div> <div>相对布局：结合RelativeLayout九宫格 表格布局和绝对布局不常用，了解就好 谷歌替代system.out.println()用Lo...</div>	<div>👁 1555</div> <div>来自： 唐三十胖子的博客</div>
<div>2018最好用百度云破解版，百度网盘不限速下载，教你如何解决百度网盘限速的方法。亲测完美使用</div> <div>百度网盘不限速 点击下载 提取码：jsk0 百度网盘不限速 点击下载 提取码：jsk0 对于大多数人来说，每次在百度网...</div>	<div>👁 98216</div> <div>来自： qq_41925894的博客</div>
<div>Proxyee-down的下载与安装教程</div> <div>Proxyee-down是monkeyWie在Github上的一个开源项目，向作者致敬。 最新版的Proxyee-down为3.12（2018.10...</div>	<div>👁 146025</div> <div>来自： shadandeajian的博客</div>
<div>Webstorm注册码 永久有效 欢迎大家使用</div>	<div>👁 9301</div>

<div>webgl第27课-三维空间-不同位置的三角形</div> <div>需要电子档书籍可以Q群：828202939 希望可以帮助和大家一起学习、一起进步！！上一节课我们学习了 颜色与纹理...</div>	<div>1973</div> <div>来自：谷子的博客</div>
<div>webgl第28课-三维空间之加入旋转矩阵的三角形</div> <div>需要电子档书籍可以Q群：828202939 希望可以帮助和大家一起学习、一起进步！！上一节课我们学习了 三维空间-...</div>	<div>1968</div> <div>来自：谷子的博客</div>
<div>微服务Springcloud超详细教程+实战（九）</div> <div>如在文档中遇到什么问题请联系作者 QQ：1172796094 本人正在找深圳Java实习工作，求大佬带飞—————...</div>	<div>14354</div> <div></div>
<div>军事理论课答案（西安交大版）</div> <div>1.1 1 【单选题】我国陆地领土面积排名世界第几？（C） A、 1 B、 2 C、 3 D、 4 2 【单选题】以下哪个国家不属于...</div>	<div>1137050</div> <div>来自：ling_wang的博客</div>
<div>Kotlin实战(三)</div> <div>Kotlin实战(三) 一、元组 1.1、二元元组 data class Pair&amp;lt;out A, out B&amp;gt;{ public val first:...</div>	<div>407</div> <div>来自：zping0808的博客</div>
<div>静态分析C语言生成函数调用关系的利器——cflow</div> <div>除了《静态分析C语言生成函数调用关系的利器——calltree》一文中介绍的calltree，我们还可以借助cflow辅助...</div>	<div>4743</div> <div>来自：方亮的专栏</div>
<div>FFmpeg详解及常用命令使用</div> <div>FFMPEG简介 FFMPEG堪称自由软件中最完备的一套多媒体支持库，它几乎实现了所有当下常见的数据封装格式、...</div>	<div>20374</div> <div>来自：qq_26464039的博客</div>
<div>安装和激活Office 2019</div> <div>有条件请支持正版！相比费尽力气找一个可能不太安全的激活工具，直接买随时随地更新的Office 365确实是最好的...</div>	<div>29144</div> <div>来自：过了即是客</div>
<div>java缓冲区</div> <div>1 缓冲区的分类 ByteBuffer CharBuffer ShortBuffer IntBuffer LongBuffer FloatBuffer DoubleBuffer 2 ByteBuffe...</div>	<div>12928</div> <div>来自：weixin_43694144的博客</div>
<div>裁员 or 缩招，AI 四起，你的工作还好吗？</div> <div>人工智能时代，AI 已经做了很多“人事”： 图片来源：中国新闻网 打扫卫生：roomba 图片来源：wirecutter.com 图...</div>	<div>9101</div> <div>来自：GitChat</div>
<div>tcp的java代码</div> <div>服务器端 package com.car.client; import java.io.IOException; import java.io.OutputStream; import java.ne...</div>	<div>12557</div> <div>来自：weixin_43694144的博客</div>
<div>webgl第36课-深度冲突-正确处理非常贴近的图形的显示问题</div> <div>需要电子档书籍可以Q群：828202939 希望可以帮助和大家一起学习、一起进步！！所有的课程源代码在我上传的资...</div>	<div>1920</div> <div>来自：谷子的博客</div>
<div>webgl第24课-几何图形的装配和光栅化</div> <div>需要电子档书籍可以Q群：828202939 希望可以帮助和大家一起学习、一起进步！！纯手打！！上一节课我们学习了 ...</div>	<div>1840</div> <div>来自：谷子的博客</div>
<div>各大磁力种子搜索引擎对比</div>	<div>443107</div> <div></div>

- vs2017安装和使用教程（详细）

VS2017如此强大，不仅仅是C语言，Python，ios，Android，Web，Node.js，Azure，Unity，JavaScript等开发都...

来自： [qq_36556893的博客](#)
- 数据结构与传统算法总结

快期末了，整理一下笔记，因为是总结嘛，比较杂的知识都没有放，都是和数据结构和传统算法相关的比较重要的...

来自： [hebtu666](#)
- webgl第34课-模型矩阵、视图矩阵、投影矩阵融合

需要电子书籍可以Q群：828202939 希望可以和大家一起学习、一起进步！！ 上一节课我们学习了 透视投影矩...

来自： [谷子的博客](#)
- webgl第35课-深度缓冲-正确处理图像的位置关系

需要电子书籍可以Q群：828202939 希望可以和大家一起学习、一起进步！！ 所有的课程源代码在我上传的资...

来自： [谷子的博客](#)



jike0901xuye

关注

原创38

粉丝27

喜欢4

评论5

等级：  访问：2万+

积分：625 排名：9万+

勋章： 



开发app



最新文章

Android JNI学习入门之使用C++链表结构存储数据

使用protobuf作为http通讯格式

Android Parcelable增加速度优化

2019人工智能薪资

Python资料免费领

会员任意学

Java薪资多少

怎样才能不被裁员

c++游戏编程

商城app开发