

# 使用Android Studio和CMake进行NDK开发 - 基础



夏日里的故事 (/u/4e9a10a9a7e6) [+ 关注](#)

2017.07.21 14:01\* 字数 1884 阅读 270 评论 0 喜欢 5

(/u/4e9a10a9a7e6)

## 1. 概述

在Android Studio 2.2之后，可以使用CMake来进行NDK开发，C/C++开发的便利性又提升了不少。这个是个好事，比较CMake使用起来还是比make要简单，并且抽象、跨平台。例如在linux可以生产linux下的makefile，在windows下可以生产Visual Studio的工程文件。

这里需要解析几个名词：

- NDK

Android Native Development Kit，里面包含各个平台上的C/C++编译器、相关头文件和库（相当于Java的库）。

- CMake

一套构建系统，类似Gradle，但是CMake不直接参与编译，而是产生其他构建系统的工程文件，再进行编译，在Android Studio当中，Gradle插件会驱动CMake产生各个平台（armeabi、armeabi-v7a、x86等）的ninja的构建文件，再驱动编译器进行编译。

- LLDB



调试器，可以用来调试原生代码，之前版本使用的是GDB。

## 2. 准备

本文是基于Android Studio 2.3来讲解的，因此需要升级到2.3。安装好2.3之后，打开【SDK Manager】

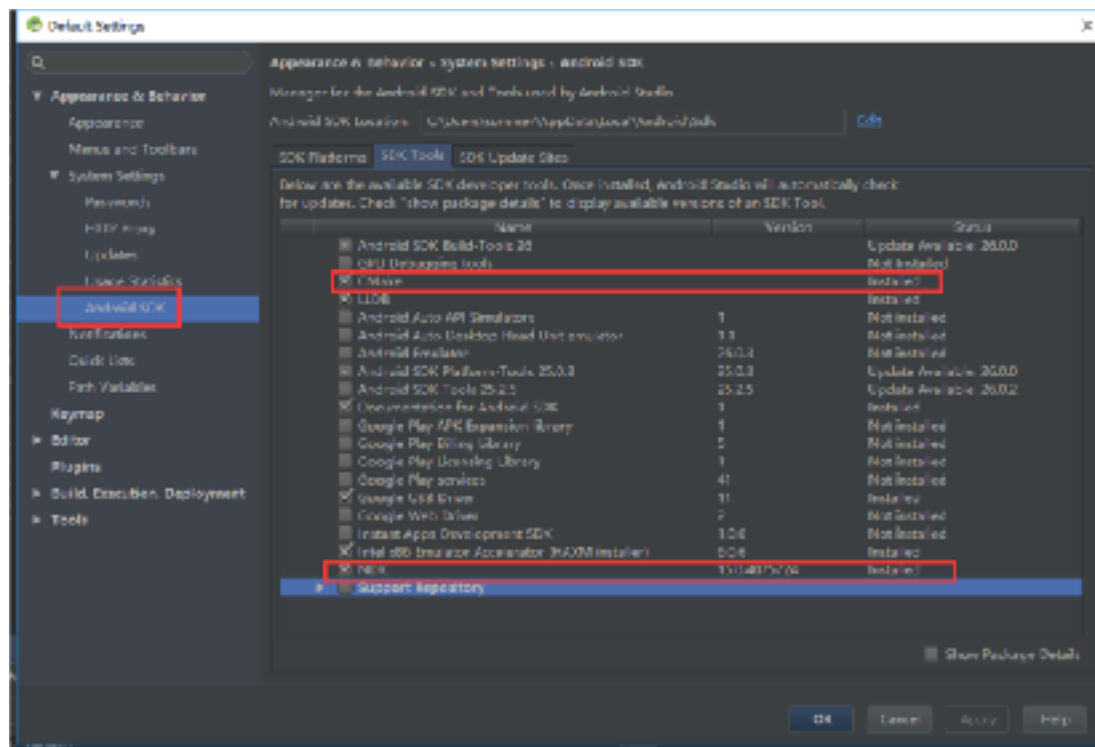


image.png

勾选：【CMake】和【NDK】两个选项，然后点击【Apply】进行安装

因为google在国内假设了镜像站点，现在不需要使用[可不描述]来更新SDK了

## 3. 实践

## 3.1 创建项目

创建项目的流程，官方文档也有：<https://developer.android.com/studio/projects/add-native-code.html> (<https://link.jianshu.com?t=https://developer.android.com/studio/projects/add-native-code.html>)

因此，我这里会在流程上补充一些说明。

新建项目，在第一页中勾选：

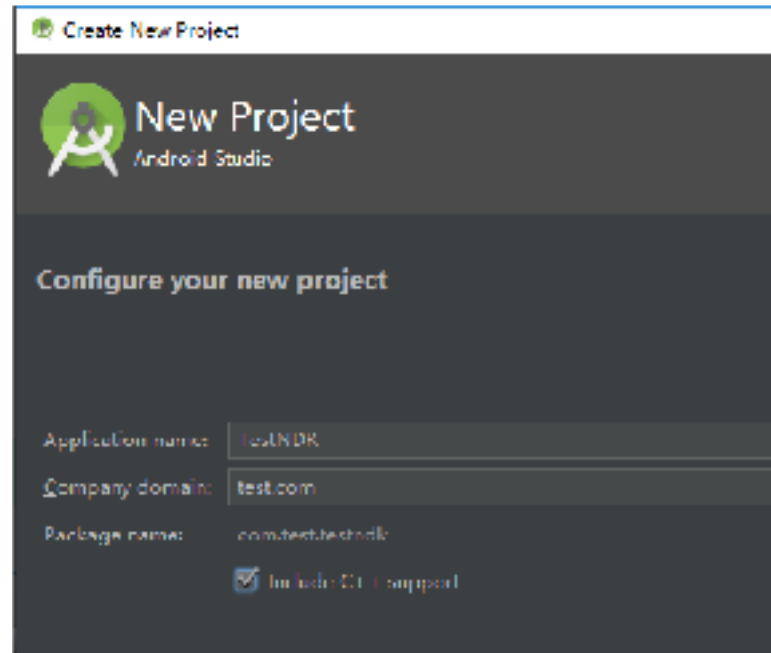


image.png

【include c++ support】来支持C++开发，如果已有项目没有勾选也没关系，可以在菜单中【link】

一路next来到最后一页，定制你的C++项目支持



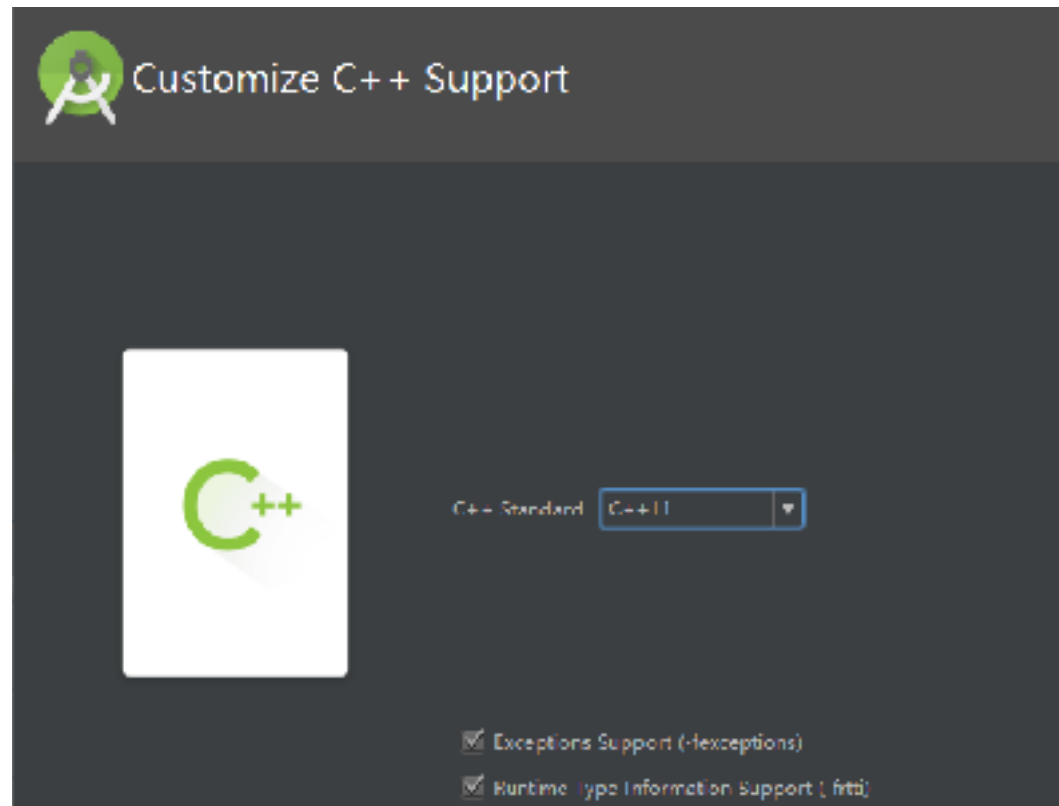


image.png

主要有三个参数组成：

- C++标准

现在基本都是C++ 11开发了，如果不是要维护非常老的代码，建议选择C++11。

C++11相对于之前的版本（C++03）增加的功能非常丰富，具体可以参考这篇文章：

<http://blog.csdn.net/zhuxianjianqi/article/details/8658169> (<https://link.jianshu.com?t=http://blog.csdn.net/zhuxianjianqi/article/details/8658169>)

- Exception Support

异常支持，如果取消掉的话，那么就不能使用 `try-catch` 进行异常处理了，建议选择。



- Runtime Type Information Support

运行时类型信息支持，在C++运行的时候，不像Java、C#等一样，可以动态获取对象的类信息，开启这个选项来支持这个功能，建议选择。

到这里，项目就创建完毕了，点击 `run` 按钮，APP就可以在模拟器或者android设备上运行。

## 3.2 工程结构

---

打开代码所在的目录，进入APP子模块，可以看到相比传统的APP项目，会多出以下文件或者目录：

- CMakeList.txt

CMake的工程文件，相当于 `build.gradle` 用于说明编译那些C/C++源码，以及相关的编译参数

- `src/main/cpp`

C/C++源码目录

- `.externalNativeBuild`

该文件夹是临时文件夹，gradle插件会调用cmake产生各个平台的临时构建文件，都存放在该目录

## 3.3 编译流程

---

需要注意的是，cmake并不能直接编译 `c/c++` 源码，需要产生 `ninja` 的项目文件，才会编译，其流程大体是这样的。



生成ninja工程 ---> 编译/链接 ---> APP打包

对于 产生ninja工程，可以通过下述三种方式：

- 修改CMakeLists.txt，然后执行 Build 菜单的 Make Project 或者 Make module 或者直接 Run
- 执行 Gradle Sync
- 执行 Build 菜单的 Refresh Linked C++ Projects

在实际使用中，有时候修改CMakeLists.txt不会重新产生ninja工程文件，导致编译会出现问题，所以官方可能也留了 Refresh Linked C++ Projects 给大家手动刷新。另外，手动添加 C/C++源码的时候，也可以通过这种方式刷新工程。

## 4. 编译参数解析

### 4.1 build.gradle

该文件可以指定工具链的大部分核心参数，里面的源码大致如下：



```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 24
    buildToolsVersion "25.0.3"
    defaultConfig {
        applicationId "com.test.ndkhelloworld"

        ...

        // 该代码块用于配置相关的参数
        externalNativeBuild {
            cmake {
                cppFlags "-std=c++11 -frtti -fexceptions"
            }
        }
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-
        }
    }

    // 该代码块用于链接到指定的CMakeLists.txt，路径是相对路径
    externalNativeBuild {
        cmake {
            path "CMakeLists.txt"
        }
    }
}

...
```

这里有两个 `externalNativeBuild` 代码块

- `android.externalNativeBuild`

`ExternalNativeBuild` 对象：<http://google.github.io/android-gradle-dsl/current/com.android.build.gradle.internal.dsl.ExternalNativeBuild.html>  
(<https://link.jianshu.com?t=http://google.github.io/android-gradle-dsl/current/com.android.build.gradle.internal.dsl.ExternalNativeBuild.html>)

主要配置 `cmake` 或者 `ndk-build` 的工程文件路径



- ``android.defaultConfig.externalNativeBuild`

`ExternalNativeBuildOptions` 对象: <http://google.github.io/android-gradle-dsl/current/com.android.build.gradle.internal.dsl.ExternalNativeBuild.html> (<https://link.jianshu.com?t=http://google.github.io/android-gradle-dsl/current/com.android.build.gradle.internal.dsl.ExternalNativeBuild.html>) 具体的参数配置。后面都是以这个配置参数来讲解。

## 4.2 CMake变量解析

可以通过 `arguments` 命令来传递CMake构建参数（这些参数，实际会传递到NDK的构建工具链），形式为：`-D参数名=参数值1 参数值2`，需要注意的是，如果有多个参数，那么必须换行来传递，例如：

```
externalNativeBuild {
    cmake {
        arguments "-DANDROID_TOOLCHAIN=gcc"
                  "-DANDROID_STL=stlport_static"
    }
}
```

以下是常用的变量：

- `ANDROID_TOOLCHAIN`

编译工具链，可选：`clang`（默认）和`gcc`（已经过期）。

- `ANDROID_PLATFORM`

android平台，例如：`android-18` 注意，该取值会影响到原生API的时候，有些原生API在低版本的android是没有的，详见：

[https://developer.android.com/ndk/guides/stable\\_apis.html](https://developer.android.com/ndk/guides/stable_apis.html) ([https://link.jianshu.com?t=https://developer.android.com/ndk/guides/stable\\_apis.html](https://link.jianshu.com?t=https://developer.android.com/ndk/guides/stable_apis.html))

- `ANDROID_STL`





STL（标准模板库）的选择，NDK自带了很多个版本的STL，功能大体上是一样的，但是授权会不一样。详细请阅读：<https://developer.android.com/ndk/guides/cpp-support.html#hr> (<https://link.jianshu.com?t=https://developer.android.com/ndk/guides/cpp-support.html#hr>)  
(`stlport`已经实现异常处理了，在低版本的NDK是不支持的)

- `ANDROID_PIE`

启用PIE（position-independent executables），如果是 `ANDROID_PLATFORM = android-16`，该选项默认开启（取值为 `ON`），否则为 `OFF`

- `ANDROID_CPP_FEATURES`

C++的功能，取值为：`rtti` 和 `exceptions`，也可以使用 `cppFlags` 指定

- `ANDROID_ALLOW_UNDEFINED_SYMBOLS`

允许未定义的符号，默认为 `FALSE`，可以取值为：`TRUE`。

- `ANDROID_ARM_MODE`

指令集模式，默认为：`thumb`，可以取值为：`arm`

- `ANDROID_ARM_NEON`

是否启用NEON，默认为`FALSE`，启用为：`TRUE`

- `ANDROID_DISABLE_FORMAT_STRING_CHECKS`

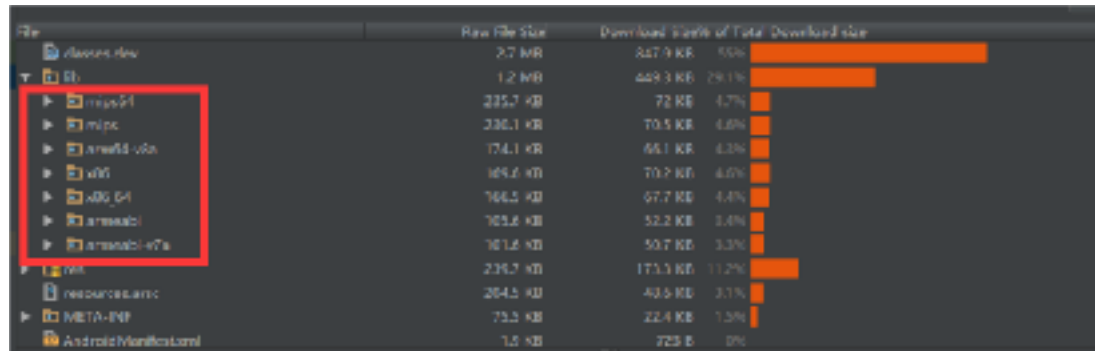
是否禁用字符串格式化检查，默认为：`FALSE`，可以取值为：`TRUE`，建议默认值，不要禁用，因为很多漏洞或者BUG都出现在字符串格式化上面。

## 4.3 abi过滤器



通过 `abiFilters` 可以编译出指定ABI的二进制文件，可选值为：`armeabi`、`armeabi-v7a`、`arm64-v8a`、`mips`、`mips64`、`x86` 以及 `x86_64`

默认情况下，编译出来的都包含上述的ABI的二进制文件，如下图：



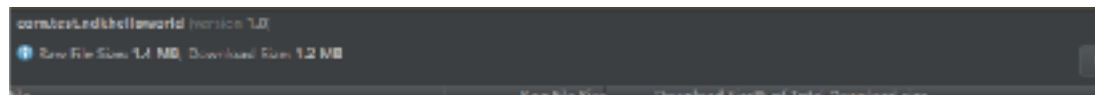
APK信息，build菜单->Analyze APK->选择产生的APK

可以清楚的看到，编译出来的二进制文件（库）可以在ARM、X86和MIPS所有平台上运行。实际上，我们想给APK瘦身的，不需要在那么多平台上运行，可以取消掉一些平台的支持，例如我们只支持armeabi和armeabi-v7a，X86和MIPS都不需要

```
externalNativeBuild {
    cmake {
        abiFilters "armeabi", "armeabi-v7a"
        cppFlags "-std=c++11 -frtti -fexceptions"
        arguments "-DANDROID_TOOLCHAIN=gcc",
                  "-DANDROID_STL=stlport_static"
    }
}
```

执行clean之后再产生APK，可以看到，只有两个ABI的二进制产生





APK信息，build菜单->Analyze APK->选择产生的APK

我发现一个问题，即使sync、clean等一系列的操作后，不会删除原有产生的ninja工程文件，可以先手动删除掉 .externalNativeBuild 目录再重试一下。

## 4.4 传递C/C++参数

cFlags和cppFlags可以传递C/C++编译参数，这里不详细讨论。

小礼物走一走，来简书关注我

赞赏支持

C++ (/nb/10652747)

举报文章 © 著作权归作者所有



夏日里的故事 (/u/4e9a10a9a7e6)

写了 16719 字，被 13 人关注，获得了 21 个喜欢

(/u/4e9a10a9a7e6)

+ 关注

喜欢 | 5



更多分享

(http://cwb.assets.jianshu.io/notes/images/13637511/weibo/image\_4





(/apps/download?utm\_source=nbc)

被以下专题收入，发现更多相似内容



Android... (/c/58b4c20abf2f?utm\_source=desktop&utm\_medium=notes-included-collection)



Android知识 (/c/3fde3b545a35?utm\_source=desktop&utm\_medium=notes-included-collection)



Android开发 (/c/d1591c322c89?utm\_source=desktop&utm\_medium=notes-included-collection)



Android... (/c/7d70f1739deb?utm\_source=desktop&utm\_medium=notes-included-collection)



Android... (/c/4688bad2bca2?utm\_source=desktop&utm\_medium=notes-included-collection)



自定义views (/c/e3e7d9e32134?utm\_source=desktop&utm\_medium=notes-included-collection)

(/p/6332418b12b1?)




utm\_campaign=maleskine&utm\_content=note&utm\_medium=seo\_notes&utm\_source=recommendation)

**Android NDK开发扫盲及最新CMake的编译使用 (/p/6332418b12b1?utm\_...**




本篇文章旨在简介 Android 中 NDK 是什么以及重点讲解最新 Android Studio 编译工具 CMake 的使用 1  
NDK 简介 在介绍 NDK 之前还是首推 Android 官方 NDK 文档。传送门 官方文档分别从以下几个方面介绍...

 Tsy远 (/u/21716b19302d?  
utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)


**Android - 收藏集 - 掘金 (/p/5ad013eb5364?utm\_campaign=maleskine&...**

用两张图告诉你，为什么你的 App 会卡顿？ - Android - 掘金 Cover 有什么料？从这篇文章中你能获得这些  
料：知道setContentView()之后发生了什么？ ... Android 获取 View 宽高的常用正确方式，避免为零 - 掘金...

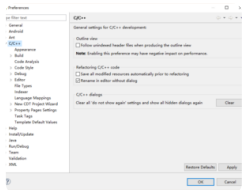
 掘金官方 (/u/5fc9b6410f4f?  
utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

**Android - 收藏集 (/p/dad51f6c9c4d?utm\_campaign=maleskine&utm\_c...**

用两张图告诉你，为什么你的 App 会卡顿？ - Android - 掘金 Cover 有什么料？从这篇文章中你能获得这些  
料：知道setContentView()之后发生了什么？ ... Android 获取 View 宽高的常用正确方式，避免为零 - 掘金...

 passiontim (/u/e946d18f163c?  
utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)


(/p/cb2f7e2da29f?



utm\_campaign=maleskine&utm\_content=note&utm\_medium=seo\_notes&utm\_source=recommendation)

**Android游戏开发实践(1)之NDK与JNI开发02 (/p/cb2f7e2da29f?utm\_cam...**

Android游戏开发实践(1)之NDK与JNI开发02 承接上篇Android游戏开发实践(1)之NDK与JNI开发01分享完JNI  
的基础和简要开发流程之后，再来分享下在Android环境下的JNI的开发，以及涉及到的NDK相关的操作。 ...


 AlphaGL (/u/4cc00e2af10b?  
utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

**JNI：为项目添加C和C++代码库 (/p/6b31c37e9fdc?utm\_campaign=male...**

developer.android.com/studio/projects/add-native-code.html 本文内容 下载 NDK 和构建工具 创建支持




C/C++ 的新项目 构建和运行示例应用 向现有项目添加 C/C++ 代码 创建新的原生源文件 创建 CMake...

 四季明 (/u/39ff43a96104?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

### 1008. 数组元素循环右移问题（另类思路） (/p/2ec403803940?utm\_campa...

一个数组A中存有N (N>0) 个整数，在不允许使用另外数组的前提下，将每个整数循环向右移M (M>=0) 个位置，即将A中的数据由 (A0 A1.....AN-1) 变换为 (AN-M ..... AN-1 A0 A1.....AN-M-1) (最后M个数...

 章斌斌 (/u/2bf44afed763?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)


(/p/a32a470263c7?)



utm\_campaign=maleskine&utm\_content=note&utm\_medium=seo\_notes&utm\_source=recommendation)

### VS code for python开发利器 (/p/a32a470263c7?utm\_campaign=malesk...


引言 最近在整理python自动化测试课程的内容，发现了微软出的vs code编辑器太牛逼了，非常好用，而且轻量的不要不要的，特此记录下，有选择纠结症的朋友我强烈推荐使用ta。 PS：兼容win10且兼容高分辨...

 测试帮日记 (/u/ff37506d06fa?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

### 我们在不同的时间段看到了同样的事 (/p/af1956400329?utm\_campaign=m...

很多时候我们会多心，因为我始终相信自己看到的，但是多在一起，多说话也许会看到一个不一样的她！我有一个闺蜜，从小学到初中是同班！初中大家有了自己的心思。而我，是一个比较细心的人，会比一般人...

 七秒记忆 (/u/5ff44b1e0a06?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

### 每天一句话 (/p/cdd46d2efc24?utm\_campaign=maleskine&utm\_content...

爱情，这东西，到底是什么呢？科学家们说是脑中的激素在搞鬼，而这激素会慢慢减少分泌，也就是说，爱情迟早要完蛋。而文学家们说，爱情这东西存在天长地久，存在永恒。我们应该相信谁呢？我相信文学...





桧枫 (/u/37a10f4dfabd?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

---

### 检讨自己吧！ (/p/1e3e652a1e95?utm\_campaign=maleskine&utm\_conte...

一件几天前听到的事情，这几天每次想起都不舒服，今天尤其愤怒，总有想去质问当事人的冲动。忍耐再三控制住了朋友圈发泄情绪的冲动，但有觉得不找个方式发泄出来会憋出内伤。一直以为自己在别人心里还...



Memory恋希 (/u/7d196316f6e0?)

utm\_campaign=maleskine&utm\_content=user&utm\_medium=seo\_notes&utm\_source=recommendation)

