

JNI

1、对 JNI 是否了解

- 参考回答：
 - Java 的优点是跨平台，但也因为其跨平台的特性导致其本地交互的能力不够强大，一些和操作系统相关的特性 Java 无法完成，于是 Java 提供 JNI 专门用于和本地代码交互，通过 JNI，用户可以调用 C、C++ 编写的本地代码
 - NDK 是 Android 所提供的的一个工具集合，通过 NDK 可以在 Android 中更加方便地通过 JNI 访问本地代码，其优点在于
 - 提高代码的安全性。由于 so 库反编译困难，因此 NDK 提高了 Android 程序的安全性
 - 可以很方便地使用目前已有的 C/C++ 开源库
 - 便于平台的移植。通过 C/C++ 实现的动态库可以很方便地在其它平台上使用
 - 提高程序在某些特定情形下的执行效率，但是并不能明显提升 Android 程序的性能

2、如何加载 NDK 库？如何在 JNI 中注册 Native 函数，有几种注册方法？

- 参考回答：

```
public class JniTest{
```

```
//加载 NDK 库
```

```
static{
```

```
    System.loadLibrary("jni-test");
```

```
}
```

```
}
```

复制代码

- 注册 JNI 函数的两种方法
 - 静态方法
 - 动态注册