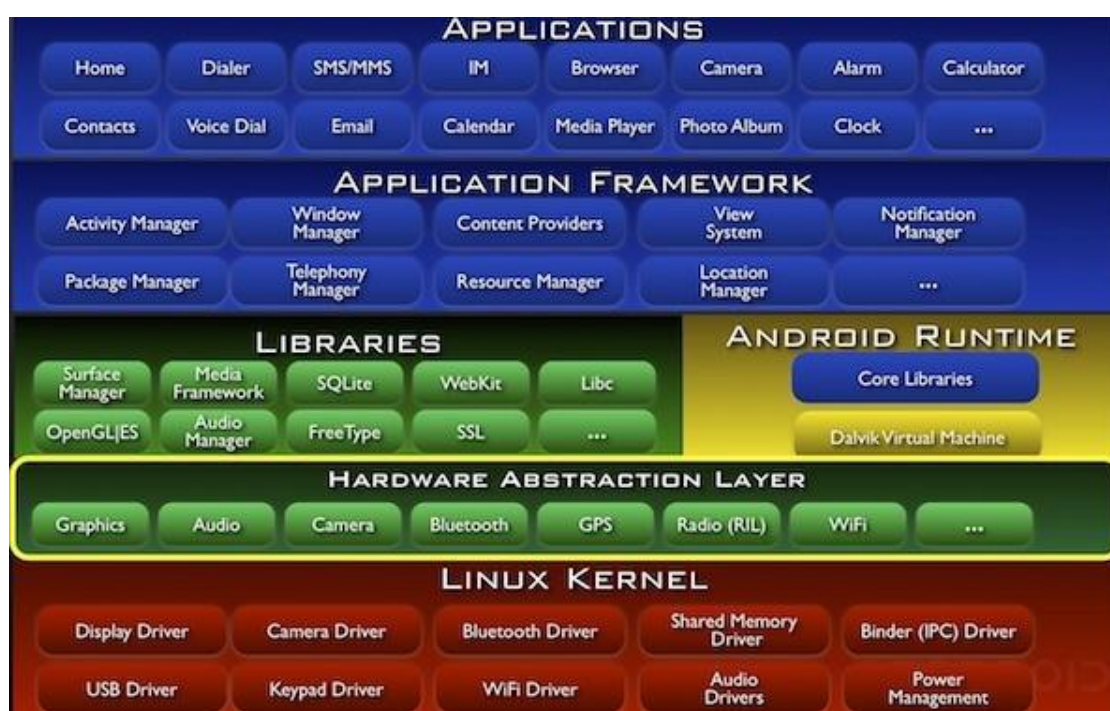


操作系统第一次作业参考解答

1、解读 Android 架构

提示：



解答：

Android 分为四个层，从高层到低层分别是应用程序层（Android 应用程序）、应用程序框架层（应用程序框架）、系统运行库层（库）和 Linux 核心层（Linux 内核）。图中蓝色部分是 JAVA 程序，黄色部分是运行 JAVA 程序的虚拟机，绿色部分是 C/C++ 程序，红色部分是 Linux 内核（C/C++ 程序）。

Android 本质就是在标准的 Linux 系统上增加 Java 虚拟机 Dalvik，并在 Dalvik 虚拟机上搭建了 JAVA 的 application framework，所有的应用程序都是基于 JAVA 的 application framework 之上。

Android 主要应用于 ARM 平台，但不仅限于 ARM，通过编译控制，在 X86、MAC 等体系结构的机器上同样可以运行。

应用程序层：

所有的应用程序都是使用 JAVA 语言编写的，每一个应用程序由一个或者多个活动组成，活动必须以 Activity 类为超类，活动类似于操作系统上的进程，但是活动比操作系统的进程要更为灵活，与进程类似的是，活动在多种状态之间进行切换。

利用 JAVA 的跨平台性质，基于 Android 框架开发的应用程序可以不用编译运行于任何一台安装有 android 系统的平台，这点正是 Android 的精髓所在。

应用程序框架层：

开发者可以完全使用核心应用程序所使用的框架 APIs，应用程序的体系结构旨在简化组件的重用，任何应用程序都能发布他的功能且任何其他应用程序可以使用这些功能（需要服从框架执行的安全限制）。这一机制允许用户替换组件。

系统运行库层：

Android 包含一个 C/C++ 库的集合，供 Android 系统的各个组件使用（例如系统 C 库，3D 库，SQLite，媒体库）

Linux 核心层：

提供核心系统服务（例如安全，内存管理，进程管理，网络堆栈，驱动模型）

2、下面的哪些指令只能在内核态中使用

a)禁止所有的中断

b)读日期-时钟时间

c)设置日期-时钟时间

改变存储器映像

提示:

常见的特权指令有以下几种:

- (1) 有关对 I/O 设备使用的指令 如启动 I/O 设备指令、测试 I/O 设备工作状态和控制 I/O 设备动作的指令等。
- (2) 有关访问程序状态的指令 如对程序状态字 (PSW) 的指令等。
- (3) 存取特殊寄存器指令 如存取中断寄存器、时钟寄存器等指令。
- (4) 其他指令

解答:

(a)(c)(d)

3、试说明库函数与系统调用的区别和联系

解答:

库函数是语言或应用程序的一部分,可以运行在用户空间中。而系统调用是操作系统的一部分,是内核提供给用户的程序接口,运行在内核空间中,而且许多库函数都会使用系统调用来实现功能。没有使用系统调用的库函数,执行效率通常比系统调用高,因为使用系统调用时,需要上下文的切换以及状态的转换(从用户态转为内核态)

4、在下图的例子中,库调用称为 read,而系统调用自身称为 read。这两者有相同的名称,哪一个更重要?

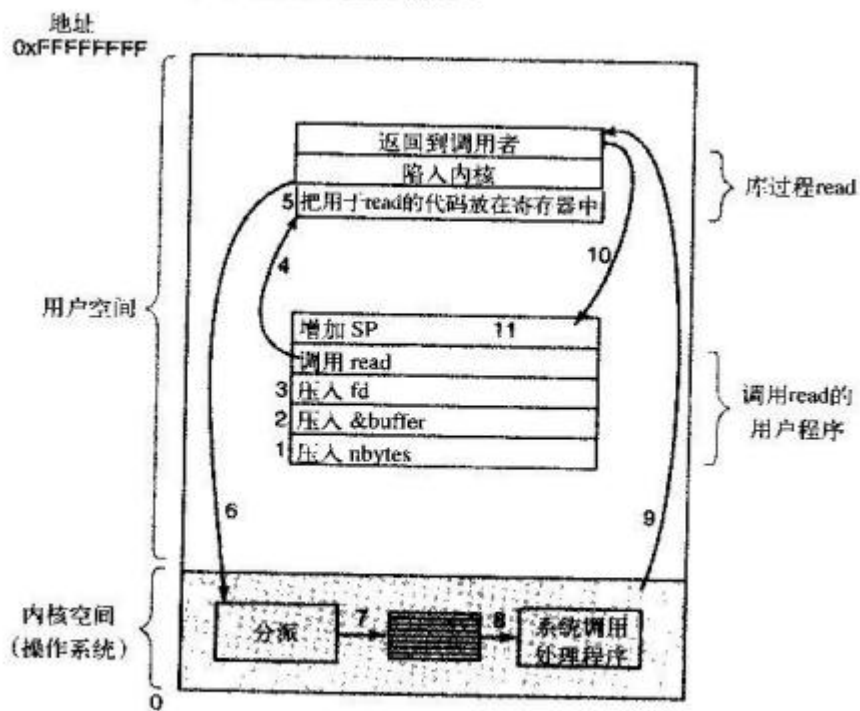


图1-17 完成系统调用read的11个步骤

解答:

系统调用实际上并没有名称，所谓的名称只是文件描述。当库调用陷入内核，它将系统调用对应序号放入寄存器或堆栈，这个序号是系统调用表的索引，整个过程没有涉及系统调用名称。另一方面，库调用的名称是十分重要的，因为这些名称是程序员使用的，出现在程序中。