1. 以下选项给出的程序中，**不**在内核态运行的是（ A ）。

A. 命令行解释程序 B. 设备驱动程序 C. 系统调用服务例程 D. 中断服务程序

1. 以下选项给出了几个在Linux系统的程序中使用的函数，其中在内核态运行的是（ B ）。

A. fread() B. sys\_read() C. read() D. \_fillbuf()

1. 假定有一个事务处理系统A，其处理器的速度为每秒钟执行5千万条指令，每个事务需要5次I/O操作，每次I/O操作需要10 000条指令。如果系统A每秒钟最多发出1000次I/O操作。则它每秒钟处理的事务数最多能达到（ A ）个。（忽略延迟并假定事务可以不受限制地并行处理）

A. 200 B. 1000 C. 2000 D. 10000

1. 在采用中断方式进行打印控制时，在打印控制接口和打印部件之间交换的信息**不包括**以下的 （ A ）。

A. 打印字符点阵信息 B. 打印控制信息 C. 打印机状态信息 D. 中断请求信号

1. 主机和外设之间的正确连接通路是（ A ）。

A. CPU和主存 ─ I/O总线 ─ I/O接口(外设控制器) ─ 通信总线(电缆) ─ 外设

B. CPU和主存 ─ I/O接口(外设控制器) ─ I/O总线 ─ 通信总线(电缆) ─ 外设

C. CPU和主存 ─ I/O总线 ─ 通信总线(电缆) ─ I/O接口(外设控制器) ─ 外设

D. CPU和主存 ─ I/O接口(外设控制器) ─ 通信总线(电缆) ─ I/O总线 ─ 外设

1. 以下是有关程序直接控制（查询）I/O方式的叙述：

① 无条件传送接口中不记录状态，无需状态查询，可直接定时访问

② 条件传送接口中有“就绪”、“完成”等状态，可定时查询或独占查询

③ 通过CPU执行相应的无条件传送程序或查询程序来完成数据传送

④ 适合于巡回检测采样系统或过程控制系统，以及非随机启动的字符型设备

以上叙述中，正确的有（ D ）。

A. 仅①和② B. 仅①和②和③ C. 仅①和③和④ D. 全部

1. 以下关于I/O子系统的描述中，**错误**的是（ D ）。

A. I/O子系统包含I/O软件和I/O硬件两大部分 √

B. I/O软件包含用户空间I/O软件部分和内核空间I/O软件部分 √

C. 内核空间I/O软件包含设备无关软件、设备驱动程序和中断服务程序 √

D. 直接控制I/O硬件的是设备驱动程序，而不是中断服务程序

1. 以下是有关对DMA方式的叙述：

① DMA控制器向CPU请求的是总线使用权

② DMA方式可用于键盘和鼠标器的数据输入

③ DMA方式下整个I/O过程完全不需要CPU介入

④ DMA方式需要用中断处理进行辅助操作

以上叙述中，**错误**的是（ B ）。

A. 仅①和② B. 仅②和③ C. 仅②和④ D. 仅③和④

简答题：

1. 有以下C语言代码：

1 #include <stdio.h>

2 int main()

3 {

4 printf( "Hello, world.\n");

5 }

假定源程序文件名为hello.c，可重定位目标文件名为hello.o，可执行目标文件名为hello，程序用GCC编译驱动程序处理，在IA-32/Linux系统中执行。回答下列问题或完成下列任务。

1. 为什么在hello.c的开头需加“#include <stdio.h>”？ 为什么hello.c中没有定义printf()函数，也没它的原型声明，但main()函数引用它时没有发生错误？
2. 加“#include <stdio.h>”是引入C的库函数，接下来直接调用C的库函数而不用自己写
3. 在编译时已经将预先编译好的C的库函数和该程序连接起来，printf()函数就是库函数，故main()函数可以直接引用
4. 需要经过哪些步骤才能在机器上执行hello程序？说明各个环节的处理过程。
5. 预处理：处理#include 预编译指令，将被包含的文件插入到该预编译指令的位置。
6. 编译：对预处理后的源程序进行编译，生成汇编语言源程序
7. 汇编：对汇编语言源程序进行汇编，生成可重定位目标文件（二进制文件）
8. 链接：将多个可重定位目标文件和标准函数库合并成为一个可执行目标文件
9. 执行：可执行目标文件保存在磁盘上，通过某种方式启动，执行hello程序
10. 为什么printf()函数中没有指定字符串的输出目的地，但执行hello程序后会在屏幕上显示字符串？

调用printf()函数最终会转到内核态，调用系统处理程序，该程序实现了将字符串输出到屏幕上

1. 字符串"Hello, world.\n"在可执行目标文件hello的哪个段中？

只读代码段

（5）若采用静态链接，则需要用到printf.o模块来解析hello.o中的外部引用符号printf，printf.o模块在哪个静态库中？静态链接后，printf.o中的代码部分（.text节）被映射到虚拟地址空间的哪个段中？若采用动态链接，则函数printf()的代码在虚拟地址空间中的何处？

libc.a

只读代码段

栈段