1、（1）一个计算个人纳税申报单的程序。

（2）“shell”程序，如C-shell或Korn shell。这样的程序必须密切监视自己的工作空间，如打开的文件、环境变量和当前工作目录。

2、当内核线程发生页面错误时，可以切换到另一个内核线程，以一种有用的方式使用交错时间。另一方面，当发生页面错误时，单线程进程将无法执行有用的工作。因此，在程序可能出现频繁的页面错误或必须等待其他系统事件的情况下，多线程解决方案甚至在单处理器系统上也能执行得更好。

3、b、c

4、由多个用户级线程组成的多线程系统不能在多处理器系统中同时使用不同的处理器。操作系统只看到一个进程，不会将该进程的不同线程调度到不同的处理器上。此时，在多处理器系统上执行多个用户级线程不会带来性能上的好处。

5、可以，系统的并行性是指在同一时间执行多个任务，并发性是指系统能够支持多个任务，允许所有任务都能取得进展。因此没有并行，通过系统的调度也可以实现并发。

6、（1）（a）加速增益 = 1 / ( 1 - 40% + 40% / 8 ) = 1.33

（b）加速增益 = 1 / ( 1 – 40% + 40% / 16 ) = 1.67

（2）（a）加速增益 = 1 / ( 1 – 67% + 67% / 2 ) = 1.5

（b）加速增益 = 1 / ( 1 – 67% + 67% / 4 ) = 1.33

（3）（a）加速增益 = 1 / ( 1 – 90% + 90% / 4 ) = 1.67

（b）加速增益 = 1 / ( 1 – 90% + 90% / 8 ) = 1.33

7、（a）此时，系统将无法利用所有可用的处理核。这可能会导致程序中某些线程需要等待其他线程完成其任务，从而减慢整体性能。此外，由于程序中有更多的用户级线程，可能会出现竞争条件和死锁等并发问题，这可能会进一步降低性能。

（b）此时，系统可以利用所有可用的处理核，从而最大化性能。由于每个内核线程都可以与一个处理核相对应，因此可以保证线程之间的负载平衡。但是，如果程序中存在竞争条件和死锁等并发问题，则性能可能会受到影响。

（c）此时，系统可以利用一些处理核来运行一部分内核线程，但无法同时利用所有处理核。这可能会导致某些内核线程需要等待其他内核线程完成其任务，从而降低性能。此外，由于有更多的用户级线程，可能会出现竞争条件和死锁等并发问题，这可能会进一步降低性能。