- **Ex. 1**

  1. POSIX is a family of standards issued by IEEE, ANSI and ISO on the implementation, application program interfaces and shell programs for UNIX operating systems.

  The idea of defining such a group of standards origins from the intension to easing cross-platform software development. Different operating systems might have differences in some ways, but as long as they comply the POSIX standard, programs written in one system will be also compatible on the other, so that the programmers only need to write code once for compability for all UNIX systems.

  The standards include: process creation/control, signal types and their control/handling, file operation, pipe operation, I/O device control(synchronous and asynchronous), scheduling, thread control, clocks and timers and the basic C library.

- **EX. 2**

  1. Because different from processes, the implementation of threads does not allow it to be interrupted and made to release memory, so it is necessary to make it release CPU voluntarily to let the other threads run, so that the pallarel running among threads can be achioeved.

  2. The biggest advantage is that the user-level thread packages is compatible in operating systems whose kernel does not support threads. Programs will be more cross-platform. The biggest disadvantage is that it is difficult to prevent a blocked thread from influencing the normal running of other threads.

  3. No because different processes does not share data among each other. So the blocking of one process will not affect the other.

  4. The program will not be compatible because there is no necessary system call functions which are supposed to be there in POSIX. So the program cannot run.