(1) How do threads reduce the high context switching overhead in processes?

Threads reduce the high context switching overhead in processes by switching only a subset of PCBs (i.e., TCB’s). This results in faster context-switching since smaller amount of data needs to be switched in thread context-switching.

(2) What information do multiple threads in a process can share?

Global variables

(3) Which of PCB (global PCB) or TCB (private PCB for each thread) do processor registers belong to? Briefly (but with a proper emphasis on the essential reason) justify your opinion.

TCB.

Processor registers belong to TCBs. It is because threads in a process usually execute different tasks (program codes). Since each thread has its own execution, a set of processor registers should belong to each thread. This, processor registers should be in TCB’s, instead of PCB’s.
(4) What are the advantages in the user-mode threads?

- Execution speed
- Program portability

(5) What are the advantages in the kernel-mode threads?

- Preemptive thread scheduling is possible
- Robust thread execution