1. **Answer:**

- Thread creation typically requires fewer resources than process creation.
- When a thread is created, it shares the resources (such as code and data) of its parent process or parent thread, so, no allocation of memory for the shared resources is necessary. Instead, only a stack and a small thread control block containing register values and other thread-related state information are required.
- For process creation, space for code and data has to be allocated for individual processes. Moreover, creating a process requires allocating space for a process control block (PCB) which is a rather large data structure.

2. **Answer:**

Any kind of sequential program is not a good candidate to be threaded.

3. **Answer:**

- In a multi-threaded solution, when a kernel-level thread is blocked (such as waiting for I/O or other system events to complete), another kernel-level thread can be switched in to continue running.
- A single-threaded process will not be capable for performing useful work when it is blocked.
- Therefore, a multi-threaded solution would perform better even on a single-processor system.

4. **Answer:**

```
Value2 = 5
Value1 = 0
```

Self-test

1. B
2. C
3. D
4. B
5. D
6. A