1. How does thread creation differ from process creation in terms of resource requirements?

2. What kind of program in which multithreading does not provide better performance than a single-threaded solution?

3. Can a multithreaded solution using multiple kernel-level threads provide better performance than a single-threaded solution on a single-processor system?

4. What would be the output from the following program?

```
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>

int   value=0;
void  *runner(void *param);

int main(int argc, char *argv[])
{
      pid_t      pid;
      pthread_t  tid;
      pthread_attr_t   attr;

      pid=fork();
      if (pid>0) {
            wait(NULL);
            printf("Value1 = %d\n", value);
      }
      else if (pid==0) {
            pthread_attr_init(&attr);
            pthread_create(&tid, &attr, runner, NULL);
            pthread_join(tid, NULL);
            printf("Value2 = %d\n", value);
      }
}

void *runner(void *param) {
      value = 5;
      pthread_exit(0);
}
```

Self-test

1. In a multithreaded environment, a _____ is defined as the unit of resource allocation and a unit of protection.
A. thread
B. process
C. string
D. trace

2. A _____ is a single execution path with an execution stack, processor state, and scheduling information.
A. domain
B. strand
C. thread
D. message

3. Which of the following is a benefit of threads?
A. Takes less time to create a new thread than a process
B. Enhances efficiency in communication
C. Takes less time to switch between threads and processes
D. All of the above

4. Which of the following states is not a state for thread?
A. Running
B. Suspend
C. Ready
D. Blocked

5. A _____ is created by invoking an application –level function in the threads library and the kernel is not aware of its existence.
A. Kernel
B. KLT
C. lightweight process
D. ULT

6. The principal disadvantage of the _____ approach is that the transfer of control from one thread to another within the same process requires a mode switch to the kernel.
A. KLT
B. API
C. VAX
D. ULT