2.31 Impliment the following C code in MIPS assembly . what is the total number of MIPS instructions needed to execute the function?

Int fib(int n){

If (n==0)

Return 0;

Else if (n==1)

Return 1;

Else

Return fib(n-1) + fib(n-2);

2.43 Write the MIPS assemble code to implement the following C code:

Lock(lk);

Shvar=max(shvar,x);

Unlock(lk);

Assume the address of the lk variable is in $a0, the address of the shvar variable is in $a1 and the value of the variable x is in $a2 your critical section should not contain any function calls. Use ll/sc instructions to implement the lock() operation, and the unlock() operation is simply an ordinary store instruction.

2.45 Using your code from exercise 2.43 as an example explain what happens when two processors begin to execute this critical section at the same time, assuming that each processor execute exactly one instruction per cycle.

A.1 Section A.5 describes how memory is portioned on most MIPS systems. Propose another way of dividing memory that meets the same goals. Explain why you would do it this way.

A.2 Is it ever safe for a user program to use the registers $k0 or $k1?