Problem 1
Consider the following parallel program:

```c
int X; // Global (shared)
void inc() { int n; for (n=0; n<20; n++) ++X; }
void dec() { int n; for (n=0; n<20; n++) --X; }
void main() { X=0; parbegin ( inc(), dec() ); print(X); }
```

a) Determine the lower and upper bounds on the final value of the shared variable \( X \) displayed by the program and explain how you arrived at this answer. Assume processes can execute at any relative speed and that a value can be incremented/decremented after it has been loaded into a register.

b) Answer Part a assuming that we generalize the program to \( N \) concurrent inc processes and \( N \) concurrent dec processes.

Problem 2
Shown below is Lamport’s bakery algorithm:

```
(1) 1 choosing[i] = TRUE;
(1) 2 num[i] = 1 + max(num[0], ..., num[N-1]);
(1) 3 choosing[i] = FALSE;
(1) 4 for j=0 to N-1 {
(1) 5 while (choosing[j]) { ... do nothing ... }
(1) 6 while ((num[j] != 0) and ((num[j], j) < (num[i], i))) {...do nothing...}
(1) 7 }
(1) 8 ... critical section ...
(1) 9 num[i] = 0;
```

The parenthesized expressions are the time units required to execute the statement once. The next column shows the line number. Furthermore, assume the following:

1) There are two processes (0, and 1) that start executing line 1 simultaneously at time 0.

2) The result of an assignment statement (e.g., line 1) is not visible until the statement finishes execution. For example, the result of line 1 is not visible until time 1.

Answer the following questions:

a) At what time will each process finish executing the last line of the algorithm? Explain how you arrived at this answer.

b) Repeat Part a but assume that there are four processes.
c) Repeat Part a assuming one change in the assumptions: Process $k$ ($k = 0, 1$) takes $10(1 - k)$ time units to execute line 2 ("$\text{num}[i] = 1 + \max(\ldots)$").

d) In general, what would be the result of deleting line 5 from Lamport’s algorithm?

**Problem 3**
Stallings, Problem 5.8