PROBLEM SAMPLE 2

Q5 (18 points)

N customers enter the bakery to buy cookies. Each customer gets its ***turn[i]***, by computing a next **number**, and waits to be served. The clerk uses a counter to keep track of the served customers. The clerk serves the customer whose ***turn[i]*** is equal to the counter. After each `serve( )`, the clerk increments the counter. When the counter reaches ***N***, the clerk considers that it is done and leaves for home.

(**Shared variables**)

turn[i] = 0 i = 1, … , N (N is initialized to 10)
number= 0;
served[i]=0;
counter = 0;

```
customer i ( ) {
number++;
turn[i] = number;
while (!served[i]){ };
getServed( );

go home;

}
```

```
clerk ( ){
while (counter < N){
        counter++;
 for( int j=1; j<=N; j++){
   if (counter ==turn[j] ){
        served[j] = True;
        serve( );  // simulated by sleep
        served[j] =False;} //if
 } // for
} // while
 leave;
} // clerk
```

All customer( ) and clerk( ) processes execute concurrently.

a) Is it possible for two customers to compute the same `number`? Explain. If yes, give the execution sequence that will show it.

b) Under the hypothesis that each customer has computed a different `number` value, is it possible for customers to compete for the same cookies (because their `turn[i]` is the same)? Explain. If yes, give the execution sequence that will show it.

c) Under the hypothesis that all customers have their `turn[i]` set before the clerk starts executing, is it possible for a customer to starve (busy wait forever)? Explain. If yes, give the execution sequence that will show it.

d) Is it possible for the clerk to never go home? Explain. If yes, give the execution sequence that will show it.

e) If there are ***N*** cookies on the shelf, is it possible for the clerk to run out of cookies before all the customers got served? Explain. If yes, give the execution sequence that will show it.