TUTORIAL FOUR

Process Synchronisation- Part I

1. Consider an e-banking system that can handle many user requests. Two different processes are responsible for handling the login and logout procedures (LoginHandler() and ExitHandler(), respectively, as shown in the below figure). A global variable N is used to record the number of users in the system and MAX is the maximum number of users allowed in the system. Describe the potential danger of this program.

```
int N=0; //the number of users
void LoginHandler( ) {
  while (N<MAX) {
     allocate resources to the user;
     N ++;
  }
}
void ExitHandler( ) {
 while (N>0) {
    deallocate resources to the user;
    N --;
  }
}
```

2. (a) Describe the three requirements that should be satisfied by a solution to the critical section problem.

   (b) The following solution attempt to solve critical section problem for two processes.

   Uses a shared variable "flag" declared as:

   ```
   int  flag[2];
   ```

   which is initialized to 0. The program for two processes are as follows:

   ```
   Process 0                          Process 1
   while (1) {                        while (1) {
       while (flag[1] == 1);              while (flag[0] == 1);
       flag[0] = 1;                       flag[1] = 1;
       critical section;                  critical section;
       flag[0] = 0;                       flag[1] = 0;
       remainder section;                 remainder section;
   };                                 };
   ```

   Determine which of the three requirements in part (a) are not satisfied. Explain your answer.

3. Mutual exclusion primitives can be implemented with busy waiting or with blocking. Discuss the applicability and relative merits of each approach.

4. Consider a computer that does not have a *TestAndSet* instruction, but does have an instruction to ***swap*** the contents of a register and memory word in a single atomic command. Show how it can be used to implement the *entry section* and *exit section* which are before and after the critical section.