CEG 4350 Midterm John Foster

1. boolean lock = false;
```
while(true){
  while(test_and_set(&lock)){   // checks for the lock
    wait(B)}                    // puts the process into the wait queue if locked + waits
  /* Critical section */
  lock=false                    // after CS, unlocks the lock
  Signal(B)                     // flips semaphore to allow the next process to enter CS
  /* Remainder section */
}
```

2. monitor TestMonitor {
```
int m = 10 // 10 processes share the file
int n = 20 // sum of process id's accessing the file can't exceed 20
condition freeProcess; // processes that can access the data
int numProcesses =0; // number of processes accessing the file
int idTotal =0; // sum of process ID's accessing the file

void checkFile(){
  if(numProcesses == m && (idTotal + process.id) > n){   // process.id is the id for the requesting process
    freeProcess.wait();
  }
  ++numProcesses;
  idTotal += process.id
}
void freeFile(){
  --numProcesses;
  idTotal -= process.id;
  freeProcess.signal();
}
```