

CS 511 – Quiz 2: Complex Atomic Actions and MEP

12 September 2018

Names:

Pledge:

Exercise 1

Consider the following method:

```
atomic compare-and-swap(common,old,new) {  
    int temp;  
    temp = common;  
    if (common==old) {  
        common = new;  
    }  
    return temp;  
}
```

Solve the MEP problem (you may disregard freedom from starvation though) using `compare-and-swap` by filling in the initialization, entry and exit protocols below. Assume that parameters are passed by reference (hence the assignment to `common` above will be seen by the caller).

```
    // >> put initialization code here  
    .  
    .  
    .  
  
thread P: {  
    while(true){  
        // non-critical section  
        // >>put entry protocol here  
        .  
        .  
        .  
        // critical section  
        // >>put exit protocol here  
        .  
        .  
        .  
        // non-critical section  
    }  
}  
  
thread Q: {  
    while(true){  
        // non-critical section  
        // >>put entry protocol here  
        .  
        .  
        .  
        // critical section  
        // >>put exit protocol here  
        .  
        .  
        .  
        // non-critical section  
    }  
}
```

Hint: Use a unique shared variable `state` that indicates the state of the critical section: 0 means no one is in the CS, 1 means thread P is in the CS and 2 means thread Q is in the CS.