**Semaphore values and purposes**:  
  
array[10-1]: This is an array of semaphores that represent a place in the first line. This prevents customers from jumping to the front of the line and starving an early process until the very end. Each customer will wait for the next in line to move forward, then they will move forward and signal the next person. The 10th person signals someone outside (first to catch the semaphore gets to come in. It is a mob out there). **Each initialized to 1.**

front: InfoDesk worker signals whoever is at the front of the line to come get their number. **Initialized to 0.**

Numbergiven: customer got their number from InfoDesk worker. **Initialized to 0.**

announcement[ number]: the announcer calls a number so that person can join the second line (the one for the agents). **All initialized to 0.**

array2[10-1]: just like the first array, but needs to be a separate list of semaphores so that one line doesn't signal the other (obviously nonsensical). **All initialized to 1.**

agentReady: Lets the customer know that an agent is ready for them. **Initialized to 4.**

givePaperkwork: starts the interaction between a customer and their agent. Allows the agent to know to start eye exam. **Initialized to 0.**

eyeExam: petitions the customer to get ready to take the exam. **Initialized to 0.**

eyeReady: customer is ready for exam. Lets agent start exam and then move to photo. **Initialized to 0.**

photo: petitions the customer to get ready for a picture. **Initialized to 0.**

photoReady: customer signals that they are ready for the photo. Allows the agent to take a photo and give a temporary license. **Initialized to 0.**

agentDone: lets customer know the temporary license is being given to them. **Initialized to 0.**

gotLicense: lets agent know customer is done and left. **Initialized to 0.**

**Pseudocode**

Customer()  
{  
 wait(array[10])  
 EnterBuilding()  
 for(i=9; i>0; i--) //This for loop keeps the line in the order they came in  
 {  
 wait(array[i])  
 signal(array[i+1])  
 }  
 signal(front)  
 wait(Numbergiven)  
 GetNumberfromInfoDesk()  
 signal(array[1])  
   
 MoveToWaitingArea()  
 wait(announcer[number])  
 for(i=9; i>0; i--)  
 {  
 wait(array2[i])  
 signal(array2[i+1])  
 }  
 wait(agentReady)  
 signal(givePaperwork)  
   
 wait(eyeExam)  
 BlinkAFewTimes()  
 signal(eyeReady)  
   
 wait(photo)  
 MakeSureHairLooksGood()  
 signal(photoReady)  
   
 wait(agentDone)  
 GetTemporaryLicense()  
 signal(gotLicense)  
 Leave()  
}

FrontDesk()  
{

while(true)  
 {  
 wait(front)  
 AssignNumberToCustomer()  
 Signal(NumberGiven)  
 }  
}

NumberAnnouncer()  
{  
 ticketNum = 0;  
 while(true)  
 {  
 wait(array2[10])  
 CallNextNumber()  
 signal(announcer[ticketNum])  
 ticketNum++;  
 }  
}

Agent()  
{  
 while(true)  
 {  
 wait(givePaperwork)  
   
 ProcessPaperwork()  
 signal(eyeExam)  
   
 wait(eyeReady)  
 GiveEyeExam()  
 signal(photo)  
   
 wait(photoReady)  
 TakePhoto()  
 GiveTempLicense()  
 signal(agentDone)  
   
 wait(gotLicense)  
   
 signal(agentReady)  
 }  
}