#define MAX\_CUST\_NO = 50;

Semaphore desk\_line = 10;

Semaphore desk = 1;

Semaphore desk\_cust\_ready = 0;

Semaphore agent\_line = 10;

Semaphore agent = 0;

Semaphore agent\_cust\_ready[2] = {0};

Semaphore mutex1 = mutex2 = mutex3 = mutex4 = mutex5 = mutex6 = 1;

Semaphore announcer[MAX\_CUST\_NO] = {0};

Semaphore desk\_finished[MAX\_CUST\_NO] = {0};

Semaphore exam [MAX\_CUST\_NO] = {0};

Semaphore finished [MAX\_CUST\_NO] = {0};

Semaphore agent\_prompt[MAX\_CUST\_NO] = {0};

Int ticket\_count;

Int ticketnr[MAX\_CUSTOMERS];

Void customer(arg) {

Int custnr = arg;

Wait(desk\_line);

Enter\_DMV();

Wait(desk);

Signal(desk\_line);

Wait(mutex2);

Enqueue1(custnr);

Signal(desk\_ready);

Signal(mutex2);

Wait(desk\_finished[custnr]);

Wait(mutex6);

Enqueue4(ticketnr[custnr]);

Signal(waiting\_room);

Signal(mutex6);

Wait(announcer[ticketnr[custnr]]);

Enter\_agent\_line();

Wait(agent);

Signal(agent\_line);

Wait(mutex5);

Dequeue3(agent\_id);

Signal(mutex5);

Wait(mutex4);

Enqueue2(custnr);

Signal(agent\_cust\_ready[agent\_id]);

Signal(mutex4);

Signal(exam[custnr]);

Wait(finished[custnr]);

Exit();

}

Void information\_desk() {

Int desk\_cust;

While(true) {

Wait(desk\_cust\_ready);

Wait(mutex2);

Dequeue1(desk\_cust\_ticket);

Signal(mutex2);

Wait(mutex3);

Ticket\_count++;

Desk\_cust\_ticketw = ticket\_count;

Signal(mutex3);

Signal(desk\_finished[desk\_cust);

Signal(desk);

}

}

Void announcer() {

Int ticketnr;

While(true) {

Wait(agent\_line);

Wait(waiting\_room);

Wait(mutex6);

Dequeue4(ticketnr);

Signal(mutex6);

Signal(announce[ticketnr]);

}

}

Void agent(arg) {

Int id = arg;

Int agent\_cust;

While(true) {

Wait(mutex1);

If(custDone == MAX\_CUST\_NO) {

Sem\_post(mutex1);

Pthread\_Exit(NULL);

}

Wait(mutex5);

Enqueue3(id);

Signal(agent);

Signal(mutex5);

Wait(agent\_cust\_ready[id]);

Wait(mutex4);

Dequeue2(agent\_cust);

Custdone++;

If(custdone == MAX\_CUST\_NO) agentDone = 1;

Signal(mutex4);

Signal(agent\_prompt[custnr]);

Wait(exam[agent\_cust]);

Signal(finished[agent\_cust]);

}

}