

```

/* program postoffice */

Semaphore customerQueue = 20;

Semaphore postalWorkerQueue = 3;

Semaphore customerReady = 0; //each customer will release this when they are ready to be
served

Semaphore scales = 1; //postalWorker will acquire when scales are needed

Semaphore addToQueue = 1; //each customer will acquire and then add their task and id to
respective queues

Semaphore getFromQueue = 1 //postalWorker will acquire and then get the customer's
information from queue

Semaphore done = 0 //postalWorker signals when they are done

Semaphore workerReady = 0; //signals when worker has added data to queue

//Data Structures needed

Queue taskList; //queue to store the randomly assigned tasks of each customer

Queue custList; // queue to store the id of each customer

Queue WorkerList; //queue to store id of the workers


Void customer()
{
semWait(customerQueue)

semWait(postalWorkerQueue)

getTask()

//get task and talk to postal worker

semWait(addToQueue)

```

```
addTaskToQueue()
addIdtoQueue()
semSignal(addToQueue)
semSignal(customerReady)
semWait(workerReady)
getWorkerId()
semWait(done)
semSignal(postalWorkerQueue)
semSignal(customerQueue)

}

void PostalWorker{
semWait(customerReady)
semWait(getFromQueue) //get customer data from queue
getFromQueue()
semSignal(getFromQueue)
addWorkerId()
semSignal(workerReady)
completeTask()
semSignal(done)
}
```

