Pseudocode

```
eating = 0
waiting = 0                    //eating and waiting keep track of the number of threads sitting at the table and waiting respectively.

mutex = Semaphore (1)
block = Semaphore (0)          //incoming customers wait on block
must_wait = FALSE              //must_wait indicates that the table is full

mutex . wait ()
if ( must_wait == TRUE || eating+1 > X )
        waiting++
        must_wait = TRUE
        mutex . signal ()
        block . wait ()

else
        eating++
        must_wait = ( waiting>0 && eating == X)
        mutex . signal ()

# Customer eats at the table

mutex . wait ()
eating--
if eating == 0
        k = min (X , waiting )
        waiting -= k
        eating += k
        must_wait = ( eating == X)
        block . signal (k)
mutex . signal ()
```