Burak Demirci

141044091   HW3 README

Ödev yapılırken kitapta bulunan aşağıdaki consumer & producer mantığı uygulanmıştır.

```
#define N 100                              /* number of slots in the buffer */
typedef int semaphore;                     /* semaphores are a special kind of int */
semaphore mutex = 1;                       /* controls access to critical region */
semaphore empty = N;                       /* counts empty buffer slots */
semaphore full = 0;                        /* counts full buffer slots */

void producer(void)
{
     int item;

     while (TRUE) {                        /* TRUE is the constant 1 */
          item = produce_item();           /* generate something to put in buffer */
          down(&empty);                    /* decrement empty count */
          down(&mutex);                    /* enter critical region */
          insert_item(item);               /* put new item in buffer */
          up(&mutex);                      /* leave critical region */
          up(&full);                       /* increment count of full slots */
     }
}


void consumer(void)
{
     int item;

     while (TRUE) {                        /* infinite loop */
          down(&full);                     /* decrement full count */
          down(&mutex);                    /* enter critical region */
          item = remove_item();            /* take item from buffer */
          up(&mutex);                      /* leave critical region */
          up(&empty);                      /* increment count of empty slots */
          consume_item(item);              /* do something with the item */
     }
}
```

Figure 2-28. The producer-consumer problem using semaphores.

Down yerine Wait sistem çağrısı uygulanmıştır

Up yerine ise Signal sistem çağrısı uygulanmıştır