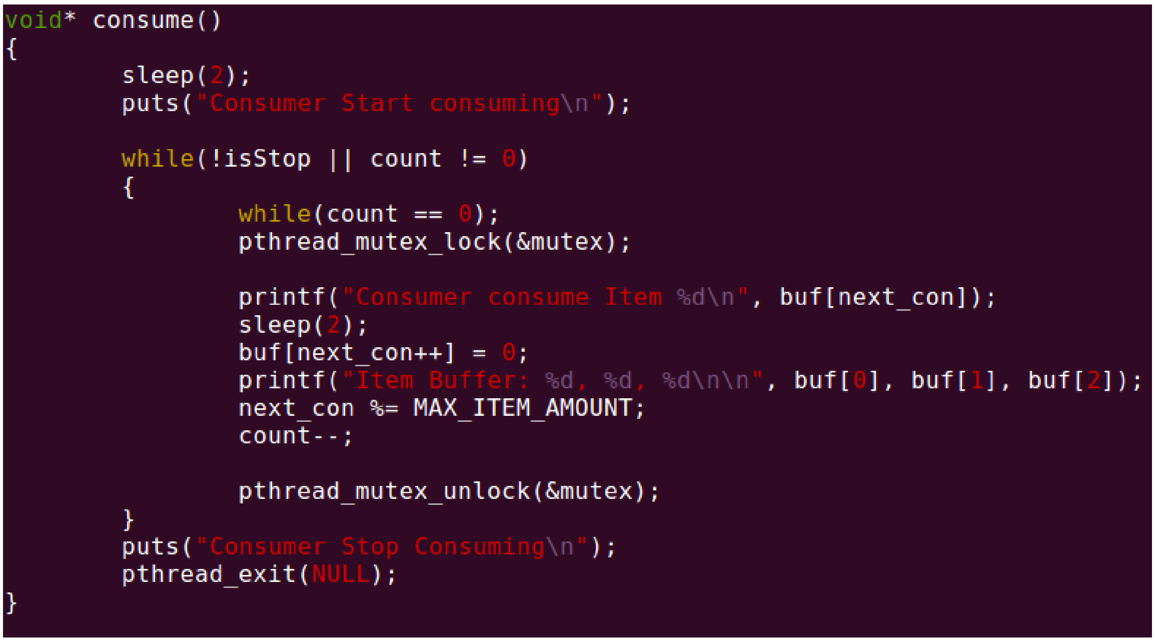
Producer Consumer project
程式說明

Consumer

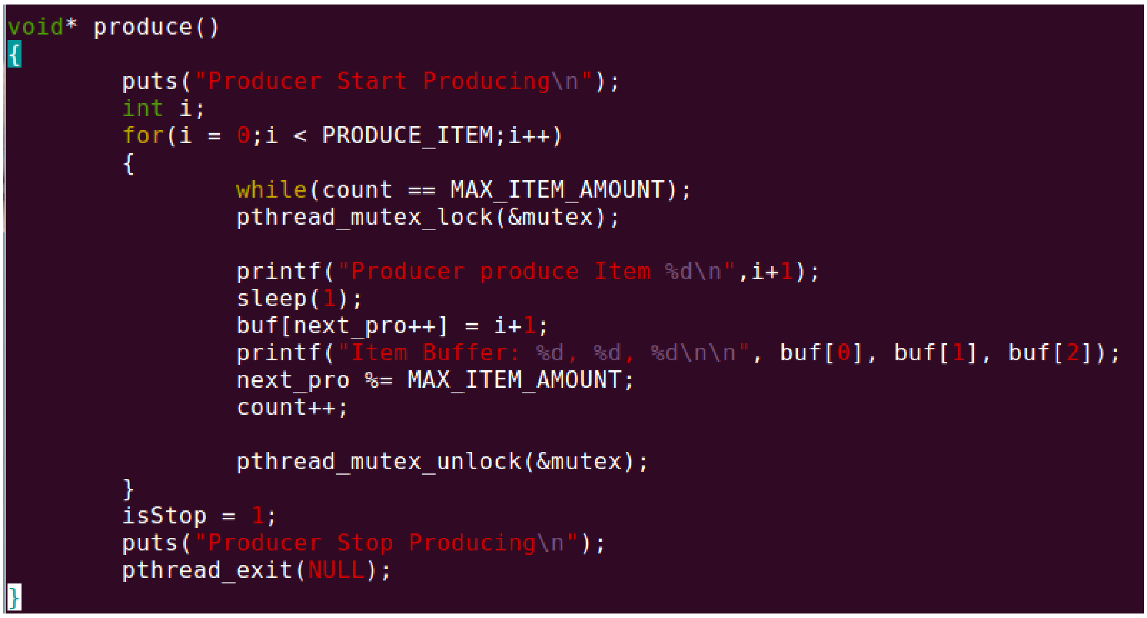
會消耗buf內的ITEM， 如果count == 0(代表buf是空的)， 則會無限迴圈等待Producer製造出新的ITEM



```c
void* consume()
{
        sleep(2);
        puts("Consumer Start consuming\n");

        while(!isStop || count != 0)
        {
                while(count == 0);
                pthread_mutex_lock(&mutex);

                printf("Consumer consume Item %d\n", buf[next_con]);
                sleep(2);
                buf[next_con++] = 0;
                printf("Item Buffer: %d, %d, %d\n\n", buf[0], buf[1], buf[2]);
                next_con %= MAX_ITEM_AMOUNT;
                count--;

                pthread_mutex_unlock(&mutex);
        }
        puts("Consumer Stop Consuming\n");
        pthread_exit(NULL);
}
```

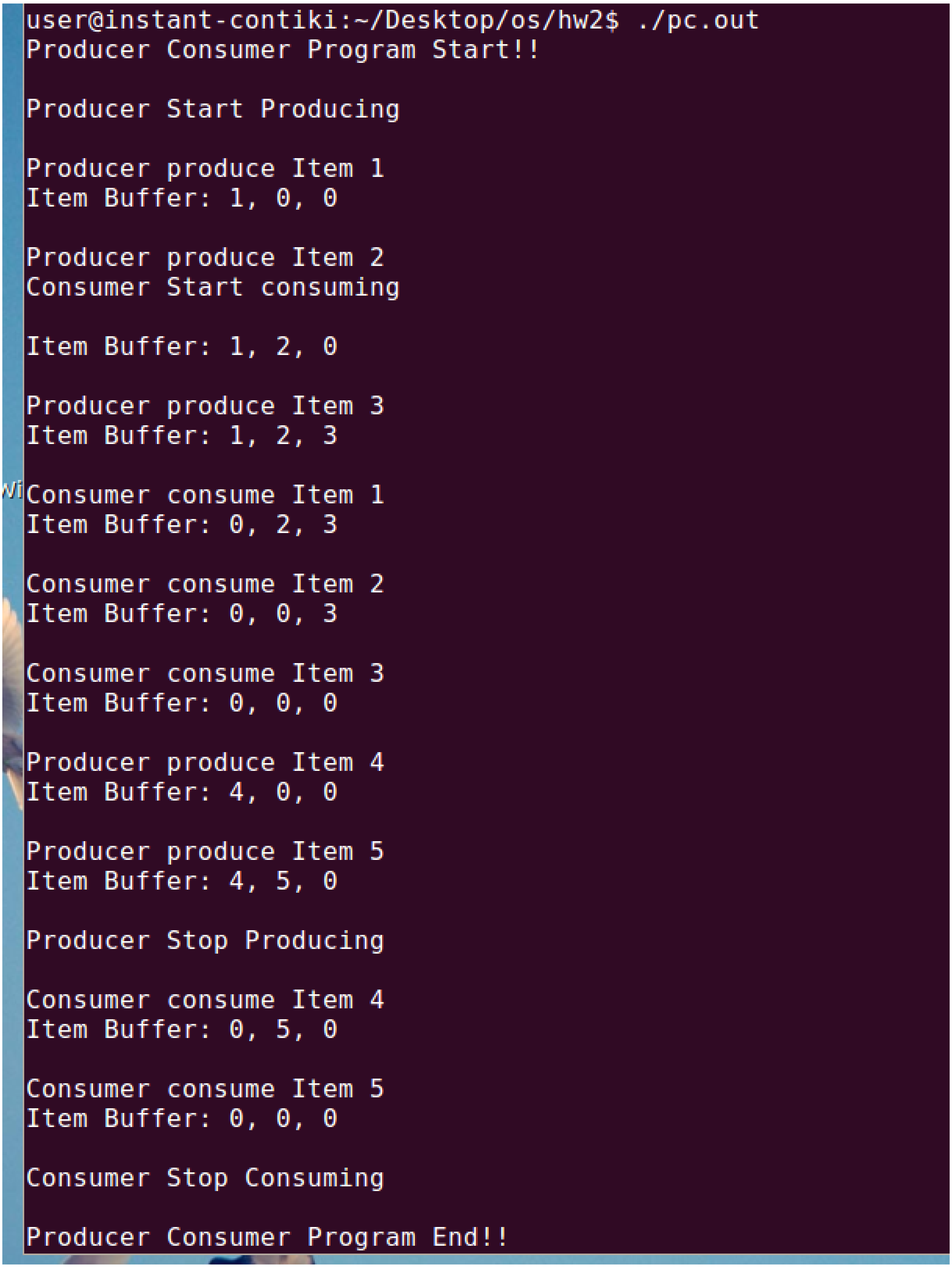
Procuder

會製造ITEM並放進buf內部， 如果count == MAX_ITEM_AMOUNT(buf滿了),則會進入無限迴圈等待consumer消耗掉ITEM。



```c
void* produce()
{
        puts("Producer Start Producing\n");
        int i;
        for(i = 0;i < PRODUCE_ITEM;i++)
        {
                while(count == MAX_ITEM_AMOUNT);
                pthread_mutex_lock(&mutex);

                printf("Producer produce Item %d\n",i+1);
                sleep(1);
                buf[next_pro++] = i+1;
                printf("Item Buffer: %d, %d, %d\n\n", buf[0], buf[1], buf[2]);
                next_pro %= MAX_ITEM_AMOUNT;
                count++;

                pthread_mutex_unlock(&mutex);
        }
        isStop = 1;
        puts("Producer Stop Producing\n");
        pthread_exit(NULL);
}
```

結果



```
user@instant-contiki:~/Desktop/os/hw2$ ./pc.out
Producer Consumer Program Start!!

Producer Start Producing

Producer produce Item 1
Item Buffer: 1, 0, 0

Producer produce Item 2
Consumer Start consuming

Item Buffer: 1, 2, 0

Producer produce Item 3
Item Buffer: 1, 2, 3

Consumer consume Item 1
Item Buffer: 0, 2, 3

Consumer consume Item 2
Item Buffer: 0, 0, 3

Consumer consume Item 3
Item Buffer: 0, 0, 0

Producer produce Item 4
Item Buffer: 4, 0, 0

Producer produce Item 5
Item Buffer: 4, 5, 0

Producer Stop Producing

Consumer consume Item 4
Item Buffer: 0, 5, 0

Consumer consume Item 5
Item Buffer: 0, 0, 0

Consumer Stop Consuming

Producer Consumer Program End!!
```