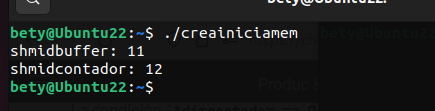
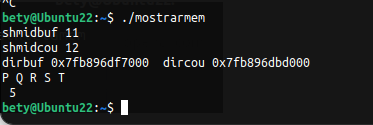
***Con las funciones de IPC ( Inter-Process Communication) creamos memoria compartida y obtenemos los identificadores. Creamos dos segmentos de memoria compartida mostramos sus identificadores en la salida.***

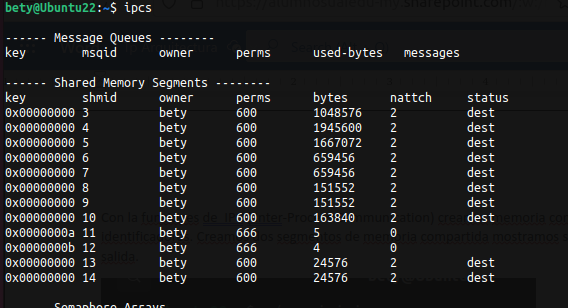


***Mostrar memoria***

***Mediante la función shmat() obtiene las direcciones de memoria compartida del búfer y el contador.***

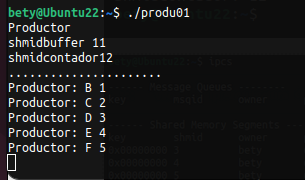
***Luego recorre el buffer e imprime los caracteres almacenados y finalmente muestra el valor del contador.***



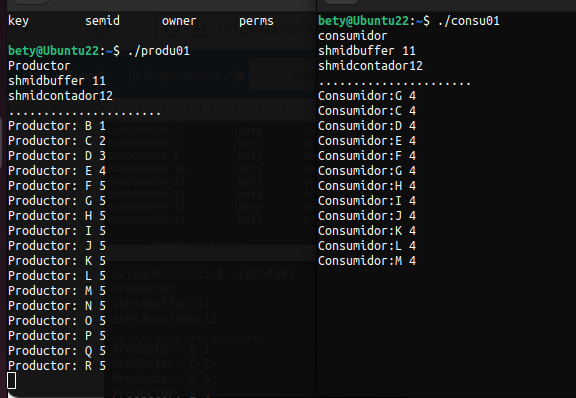




***Ejecutamos productor versión 01 que se bloquea por buffer lleno.***



***Ejecutamos Consumidor versión 01 al mismo tiempo que el productor***



***“En la versión con espera ocupada o activa, tanto el productor como el consumidor revisan constantemente el estado del buffer y esperan hasta que las condiciones necesarias se cumplan para realizar sus operaciones.*** ***Puede ser ineficiente en términos de uso de recursos del sistema, ya que los procesos consumen tiempo de CPU incluso cuando no pueden realizar ninguna operación***

***El productor y el consumidor comparten un área de memoria para el buffer y otra para el contador.***

***El productor produce elementos y los almacena en el buffer hasta llenarlo (5 bytes) queda en estado de espera ocupada.***

***EL consumidor si el buffer está vacío, espera activamente (realiza un bucle mientras espera a que haya elementos disponibles en el buffer).***

***El productor incrementa el contador después de producir un elemento y el consumidor lo decrementa después de consumir un elemento.***

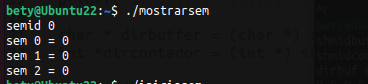
***Versión 02***

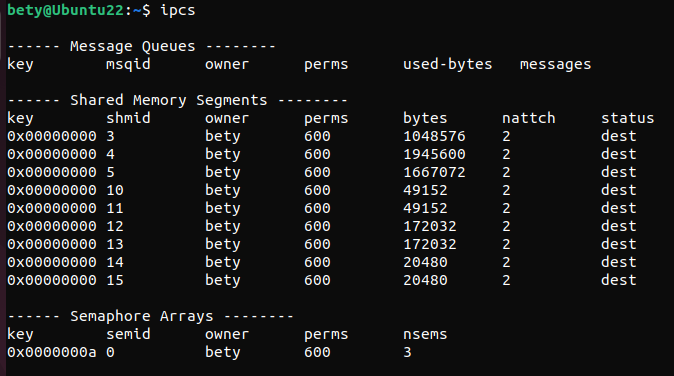
***Con la función ´semget()´ creamos un conjunto de semáforos, en este caso 3.***

***Sem 0= semáforo binario de mutua exclusión***

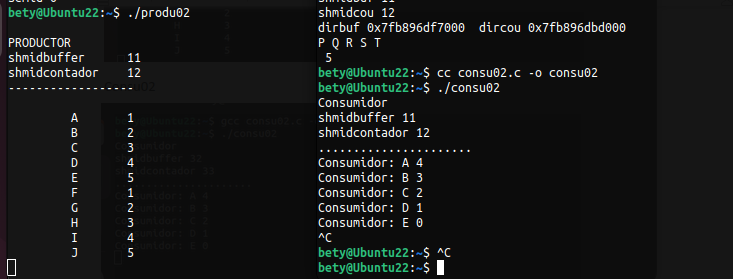
***Sem1= semáforo contador para productor (reemplazo de la espera ocupada)***

***Sem2= semáforo contador para consumidor (reemplazo de la espera ocupada)***



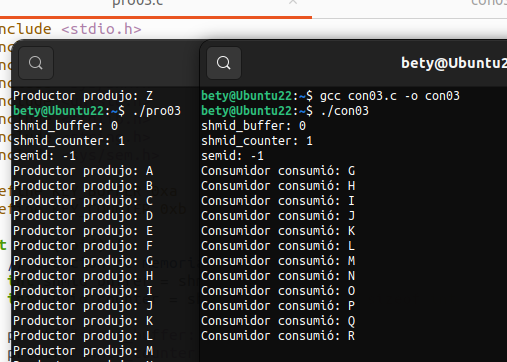




***Sem1= semáforo de exclusión mutua*** 

***El productor realiza una operación de espera en el semáforo de exclusión mutua para adquirir acceso exclusivo a la sección crítica (el buffer). Verifica si el buffer está lleno constantemente utilizando tiempo de CPU, si el buffer está lleno, libera el semáforo de exclusión mutua y espera activamente (realiza un bucle mientras espera a que haya espacio disponible en el buffer, una vez que hay espacio disponible, agrega un elemento al buffer, luego libera el semáforo para que el consumidor acceda a la sección critica.***

***En esta versión del semáforo de exclusión mutua aún se utiliza la espera activa para verificar la condición de buffer lleno o vacío, lo que puede ser ineficiente y consumir excesivo tiempo de CPU.***



***Versión Sin espera ocupada o espera activa, utilizando el semáforo de exclusión mutua y semáforos contadores.***

***En esta versión, se utiliza el semáforo de exclusión mutua para proteger la sección crítica y evitar condiciones de carrera. Además, se utiliza el semáforo como contador para controlar la disponibilidad de elementos en el buffer, eliminando la necesidad de espera activa. Los procesos realizan operaciones de espera y señal en los semáforos correspondientes para sincronizarse y evitar conflictos en el acceso al buffer compartido.***