```
/*producer consumer problem using semaphores*/
#include<stdio.h>
#include<stdlib.h>

int mutex = 1, full = 0, empty = 3, x = 0;

int wait(int s);
int signal(int s);
void producer();
void consumer();

int main() {
   int n;
   printf("MAIN MENU\n");
   printf("1. producer\n2. consumer\n3. exit\n");

   while (1) {
      printf("\nEnter your choice: ");
      scanf("%d", &n);
      printf("\n");

      switch (n) {
      case 1:
         if ((mutex == 1) && (empty != 0))
            producer();
         else
            printf("Buffer is full\n");
         break;

      case 2:
         if ((mutex == 1) && (full != 0))
```

```
            consumer();
        else
            printf("Buffer is empty\n");
        break;

    case 3:
        exit(0);

    default:
        printf("INVALID CHOICE\n");
        break;
    }
  }
  return 0;
}

int wait(int s) {
  return (--s);
}

int signal(int s) {
  return (++s);
}

void producer() {
  mutex = wait(mutex);
  full = signal(full);
  empty = wait(empty);
  x++;
  printf("\nProducer produces the item: %d", x);
  mutex = signal(mutex);
```

```
}


void consumer() {
    mutex = wait(mutex);
    full = wait(full);
    empty = signal(empty);
    printf("\nConsumer consumes item: %d", x);
    x--;
    mutex = signal(mutex);
}
```