

# COMPUTACIÓN CONCURRENTE

## EXAMEN 3: Algoritmos sin bloqueo y Consenso

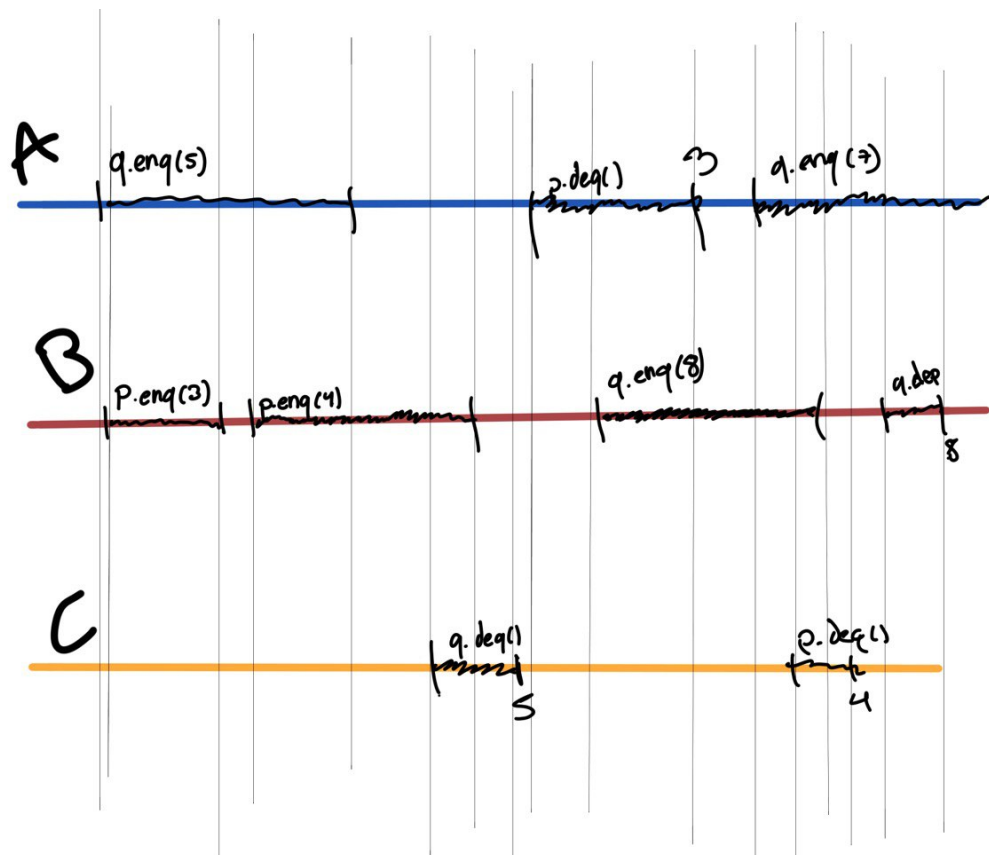
Prof. Manuel Alcántara Juárez  
manuelalcantara52@ciencias.unam.mx

Fecha: 2 de Febrero de 2021

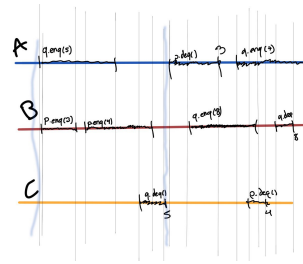
1. Considerando la siguiente historia  $H$ :

```
<A, q.enq(5)>
<B, p.enq(3)>
<B, p:void>
<B, p.enq(4)>
<A, q:void>
<C, q.deq()>
<B, p:void>
<C, q:5>
<A, p.deq()>
<B, q.enq(8)>
<A, p:3>
<A, q.enq(7)>
<C, p:deq()>
<B, q:void>
<C, p:4>
<B, q.deq()>
<B, q:8>
```

- a) **[0.5pt]** Dibuja el diagrama que representa la ejecución asociada.



b) [0.5pt] Marca los espacios de quietud, así como los puntos de linearización.



c) [0.5pt] ¿Qué es  $H|B$ ?

Es el subconjunto de  $H$ , o subhistoria de  $H$  que solo incluye a los eventos de  $B$ , en este caso sería:

$\langle B, p.enq(3) \rangle$   
 $\langle B, p:void \rangle$   
 $\langle B, p.enq(4) \rangle$   
 $\langle B, p:void \rangle$   
 $\langle B, q.enq(8) \rangle$   
 $\langle B, q:void \rangle$   
 $\langle B, q.deq() \rangle$   
 $\langle B, q:8 \rangle$

d) [0.5pt] ¿Que instrucción agregarías o quitarías para que  $H$  sea completa? La forma mas facil es simplemente quitar la ultima instrucción de  $A$ , es decir dejar:

$\langle A, q.enq(5) \rangle$   
 $\langle B, p.enq(3) \rangle$   
 $\langle B, p:void \rangle$   
 $\langle B, p.enq(4) \rangle$   
 $\langle A, q:void \rangle$   
 $\langle C, q.deq() \rangle$   
 $\langle B, p:void \rangle$   
 $\langle C, q:5 \rangle$   
 $\langle A, p.deq() \rangle$   
 $\langle B, q.enq(8) \rangle$   
 $\langle A, p:3 \rangle$   
 $\langle C, p:deq() \rangle$   
 $\langle B, q:void \rangle$   
 $\langle C, p:4 \rangle$   
 $\langle B, q.deq() \rangle$   
 $\langle B, q:8 \rangle$

Pero tambien perfectamente puede agregar al final esto:

$\langle A, q.enq(5) \rangle$   
 $\langle B, p.enq(3) \rangle$   
 $\langle B, p:void \rangle$   
 $\langle B, p.enq(4) \rangle$   
 $\langle A, q:void \rangle$

```

<C, q.deq()>
<B, p:void>
<C, q:5>
<A, p.deq()>
<B, q.enq(8)>
<A, p:3>
<A, q.enq(7)>
<C, p:deq()>
<B, q:void>
<C, p:4>
<B, q.deq()>
<B, q:8>
<A, q:void>

```

e) [0.5pt] ¿Es  $H$  bien formada? ¿Porqué?

Que  $H$  este bien formada quiere decir que todas las subhistorias, en este caso  $(H|A, H|B, H|C)$ , son secuenciales. Y es cierto:

```

<A, q.enq(5)>
<A, q:void>
<A, p.deq()>
<A, p:3>
<A, q.enq(7)>

```

```

<B, p.enq(3)>
<B, p:void>
<B, p.enq(4)>
<B, p:void>
<B, q.enq(8)>
<B, q:void>
<B, q.deq()>
<B, q:8>

```

```

<C, q.deq()>
<C, q:5>
<C, p:deq()>
<C, p:4>

```

f) [1.0pt] Indica todas las relaciones de precedencia en  $H$ .

Facil de hacerse pero muy tardado de escribir, no me dio tiempo de hacerlo, perdon :c

g) [2.0pt] Construye una historia legal y secuencial  $S$  equivalente a  $H$ . Argumenta porqué es legal y equivalente. Esta es compleja de explicar de manera formal porque es equivalente y legal, es trivial demostrar que es secuencial, pues después de cada invocación lo que sigue es justo el retorno, ahora podemos decir que es una historia legal porque no estamos moviendo de lugar los eventos, es podemos demostrar que son equivalente pues no  $S|A = H|A$  y  $S|B = H|B$  y  $S|C = H|C$ .

<A, q.enq(5)>  
<A, q:void>  
<B, p.enq(3)>  
<B, p:void>  
<C, q.deq()>  
<C, q:5>  
<B, p.enq(4)>  
<B, p:void>  
<A, p.deq()>  
<A, p:3>  
<B, q.enq(8)>  
<B, q:void>  
<C, p.deq()>  
<C, p:4>  
<B, q.deq()>  
<B, q:8>  
<A, q.enq(7)>

2. [1.5pt] Para cada una de las siguientes historias, menciona si son secuencialmente consistentes, tienen la propiedad de consistencia de quietud o son linearizables. Justifica tu respuesta.

a)  $\langle A, q.enq(x) \rangle$   
 $\langle B, q.enq(y) \rangle$   
 $\langle A, q.void \rangle$   
 $\langle B, q.void \rangle$   
 $\langle B, q.deq() \rangle$   
 $\langle A, q.deq() \rangle$   
 $\langle B, q.x \rangle$

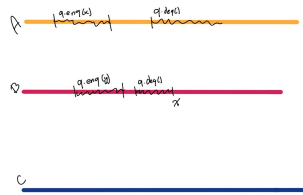


Figura 1: El diagrama sería

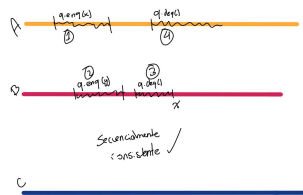


Figura 2: Secuencialmente consistente, pues podemos darle un orden a los eventos tal que el orden de los eventos dentro de cada hilo se mantiene y el orden resultante tiene sentido.

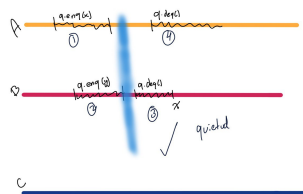


Figura 3: Consistencia de quietud, pues solo existe un espacio de quietud, y podemos dar un orden, el que mostramos por ejemplo, donde se respeta el orden de los eventos por bloques de quietud y el orden resultante tiene sentido.

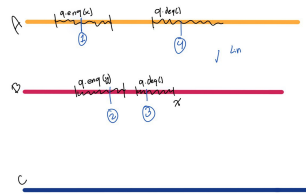


Figura 4: Linearizable, pues podemos dar un tiempo puntual para cada evento y con ello un orden resultante que tiene sentido.

b)  $\langle A, q.\text{enq}(x) \rangle$   
 $\langle B, q.\text{enq}(y) \rangle$   
 $\langle A, q:\text{void} \rangle$   
 $\langle B, q:\text{void} \rangle$   
 $\langle A, q.\text{deq}() \rangle$   
 $\langle C, q.\text{deq}() \rangle$   
 $\langle A, q:y \rangle$   
 $\langle C, q:y \rangle$

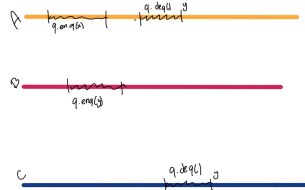
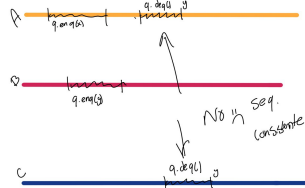
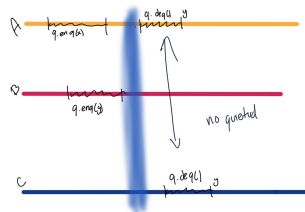


Figura 5: El diagrama sería

Figura 6: No es secuencialmente consistente, pues no importa el orden que pongamos non existen dos eventos de  $\text{enq}(y)$  que nos permitan explicar los 2  $\text{deq}()$  que dan yFigura 7: No hay consistencia de quietud, pues no importa el orden que pongamos non existen dos eventos de  $\text{enq}(y)$  que nos permitan explicar los 2  $\text{deq}()$  que dan y



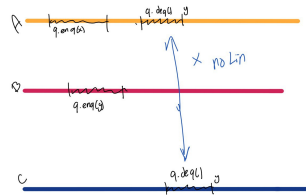


Figura 8: No es linearizable, pues no importa el orden que pongamos no existen dos eventos de  $enq(y)$  que nos permitan explicar los 2  $deq()$  que dan y

c) <A, s.push(10)>  
 <C, s.tryPop()>  
 <B, s.tryPop()>  
 <A, s:void>  
 <A, s.push(20)>  
 <B, s:10>  
 <A, s:void>  
 <C, s.emptyException>

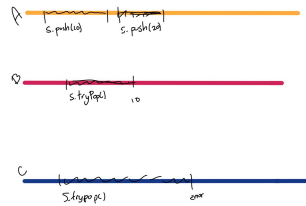


Figura 9: El diagrama sería

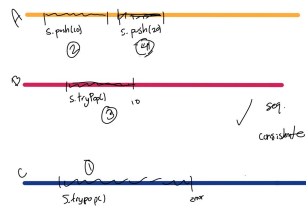


Figura 10: Secuencialmente consistente, pues podemos darle un orden a los eventos tal que el orden de los eventos dentro de cada hilo se mantiene y el orden resultante tiene sentido.

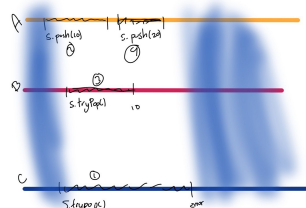


Figura 11: Consistencia de quietud, pues solo existe un espacio de quietud, y podemos dar un orden, el que mostramos por ejemplo, donde se respeta el orden de los eventos por bloques de quietud y el orden resultante tiene sentido, esto es mas facil pues todos los eventos ocurren en el mismo bloque.

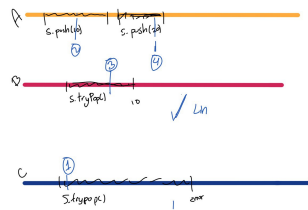


Figura 12: Linearizable, pues podemos dar un tiempo puntual para cada evento y con ello un orden resultante que tiene sentido.

3. [1.0pt] Da un ejemplo de una ejecución que tenga consistencia de quietud pero no secuencial; y otra que tenga consistencia secuencial pero no de quietud. Argumenta porque las ejecuciones no cumplen con cada propiedad

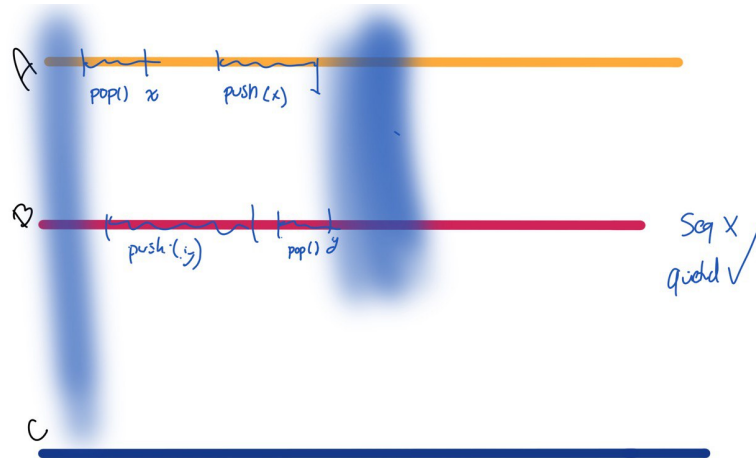


Figura 13: La idea es sencilla, es fácil acomodar los eventos para mostrar la consistencia de quietud, pues al estar en el mismo bloque podemos poner primero el push del hilo a, luego el pop del hilo a, luego el push del hilo b y luego el pop del hilo b, pero debido a que en la consistencia secuencial no podemos mover el orden de los eventos de un mismo hilo es imposible lograr un dato de un pop que solo puede provenir de un push posterior

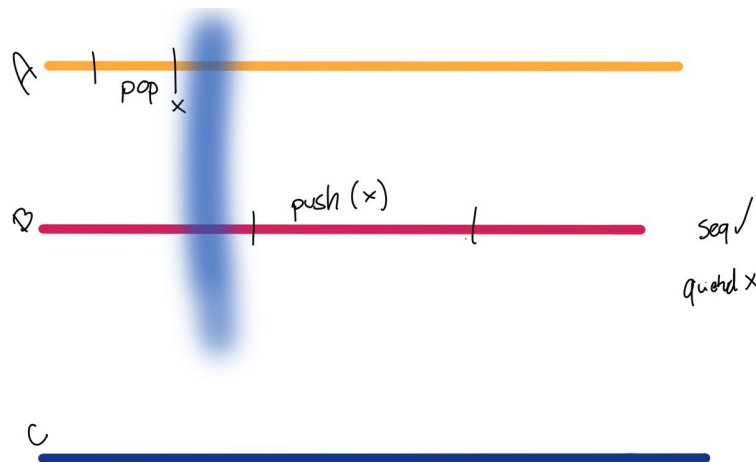


Figura 14: La idea es poner un espacio de quietud que obligue a pensar en que primero se hizo un pop, lo cual es imposible, pero es trivial resolver el problema si buscamos consistencia secuencial, pues podemos mover el orden y decir que el evento del hilo b se ejecuto primero, y al final el hilo a

## 4. [2.0pt]

Demuestra que la siguiente implementación no es linearizable para más de 3 hilos:

```
public void enq(T x) {  
    if (tail - head == capacity)  
        throw FullException();  
    items[tail % capacity] = x;  
    tail++;  
}  
  
public T deq() {  
    if (tail - head == 0)  
        throw EmptyException();  
    T x = items[head % capacity];  
    head++;  
    return x;  
}
```