

TAREA

Entrega: Viernes 26/11/2004

El problema de la cena de los filósofos

1. Había una vez cinco filósofos que vivían juntos. La vida de cada filósofo consistía principalmente en pensar y comer y, tras años de pensar, todos los filósofos se habían puesto de acuerdo en que la única comida que contribuía a sus esfuerzos era el arroz.

Los preparativos de la comida eran simples: una mesa redonda en la que había cinco platos de arroz, uno para cada filósofo, y cinco palillos. Un filósofo que quisiera comer iría a su lugar asignado en la mesa y, usando los dos palillos de cada lado del plato, se comería el arroz. El problema es el siguiente: inventar un ritual (algoritmo) que permita comer a los filósofos. El algoritmo debe satisfacer la exclusión mutua (dos filósofos no pueden emplear el mismo palillo a la vez), además de evitar el interbloqueo y la inanición (en este caso, este último término tiene un significado literal además del algorítmico).

Una primera solución al problema de la cena de los filósofos es:

```
/* programa cena_filósofos */
semáforo palillo[5] = {1};
int i;
void filosofo(int i)
{
    while (cierto)
    {
        pensar ( );
        wait (palillo [i]);
        wait (palillo [(i + 1) mod 5]);
        comer ( );
        signal (palillo [(i + 1) mod 5]);
        signal (palillo [i]);
    }
}
void main ( )
{
    parbegin (filosofo (0), filosofo (1), filosofo (2), filosofo (3), filosofo (4));
}
```

Este algoritmo sugiere una solución por medio de semáforos. Cada filósofo toma primero el palillo de su izquierda, y después el de su derecha. Cuando un filósofo termina de comer, devuelve los dos palillos a la mesa. Esta solución, desafortunadamente, produce interbloqueo: si todos los filósofos están hambrientos al mismo tiempo, todos se sientan, todos toman el palillo de su izquierda y todos intentan tomar el otro palillo que no estará. Esta solución es poco decorosa, pues todos los filósofos pasan hambre.

Para superar el riesgo de interbloqueo se podrían adquirir 5 palillos adicionales (una solución más saludable), o enseñar a los filósofos a comer arroz con un solo palillo. Su misión: proponer una modificación a este algoritmo libre de interbloqueo e inanición.

Nota: No hagan caso de la solución de los libros, pues alguna de ellas puede conducir a inanición.

2. Cenicienta y el Príncipe se quieren divorciar. Para dividir sus propiedades, han acordado el siguiente algoritmo. Cada mañana uno de ellos debe enviar una carta al abogado del otro para solicitar un elemento de su propiedad. Puesto que una carta tarda un día en ser entregada, han acordado que si ambos descubren que han solicitado el mismo artículo el mismo día, al día siguiente enviarán una carta para cancelar la solicitud. Entre sus propiedades están su perro Wooper, su perrera, su canario Piolín y la jaula del canario. Los animales aman sus casas, por lo cual han acordado invalidar cualquier división de la propiedad que separe a un animal de su casa, por lo que la división deberá volver a iniciarse desde cero. Tanto Cenicienta como el príncipe desean de forma desesperada a Wooper. Ambos se van de vacaciones (separados) y cada uno de ellos programa una computadora personal para manejar la negociación. Al regresar de sus vacaciones, las computadoras continúan negociando. ¿Por qué? ¿Es posible el bloqueo? ¿Es posible la inanición? Explique.