Filósofos comensales

Se trata de filósofos que dedican su vida a pensar y después comer

Deben compartir cubiertos, y necesitan dos tenedores para comer

* Procesos:
  + Filósofos
* Elementos compartidos
  + Número tenedores

Los filósofos necesitan tenedor, y no se van a ir sin él. Los tenedores entonces podrían ser semáforos en vez de una variable compartida.

* Semáforos (Serán cinco iniciados a uno contadores de recursos, uno por cada tenedor, por lo que todos estarán inicializados a 1)

Proceso filósofo:

While (1){

Pensar();

Wait(tenedor[i]);

Wait(tenedor[(i+1)%5]); //Módulo cinco pues como son solo cinco, cada vez que llegue a este pase a 0 de nuevo

Comer();

signal(tenedor[i]);

signal(tenedor[(i+1)%5]);

}

Podríamos poner otro semáforo que represetnase un sirviente, inicializado a 4, solo podrñia permitir que hubiese cuatro comensales para que nadie quede esperando infiitamente:

While (1){

Wait(sirviente);

Pensar();

Wait(tenedor[i]);

Wait(tenedor[(i+1)%5]); //Módulo cinco pues como son solo cinco, cada vez que llegue a este pase a 0 de nuevo

Comer();

signal(tenedor[i]);

signal(tenedor[(i+1)%5]);

signal(sirviente);

}

Dejando todos los tenedores en una bandeja en medio:

While (1){

Pensar();

Wait(Par\_tenedores); //se decrementa la variable en dos

Comer();

signal(Par\_tenedores);

}

**Solución correcta:**