

ネットワークプログラミング

Jacques Garrigue, 2005 年 1 月 18 日

4 ペトリ・ネットの応用

4.1 哲学者の晩餐

こんな有名な並列プログラミングの問題がある。

晩餐会に哲学者が N 人集ったが、残念ながら箸は N 本しかない。丸い机に座って、箸は二人の哲学者の間に一本ずつ置いてある。はたして、彼等は全員食事ができるのか。

Dijkstra が考えたもので、デッドロックや不公平が起こりうる。

各哲学者は次の動作をくりかえす

- 両側の箸を取る
- しばらく食べる
- 箸を置く
- しばらく考える

それをペトリネットで表現するために、二人程度にするが、通常は 5 人の例を考える。

実習

UDP のクライアントとサーバを元にして、晩餐会を実装してください。サーバはテーブルを表して、クライアントは各哲学者を表している。以下のようなプロトコルで通信を行う。

1. まず、サーバを立ち上げる。参加可能な哲学者の数はあらかじめ決っている。(3 など)

2. 新しいクライアントが参加するときは、まずサーバから席をもらう。SEATと書いてあるUDPのメッセージを送り、番号を返してもらう。
3. 食べたいときは自分の左と右の箸を取らないといけない。自分の番号が k なら、左はGET k 、右はGET $k + 1$ ($k = N$ ならば GET 1) で取ろうとする。取ることができれば、サーバがOKを返すが、既にとり隣の哲学者が使っていれば、NOを返す。
4. 食べ終わったら、PUT k 、PUT $k + 1$ で箸を返す。返事はOK。
5. しばらく考えると、また食べようとする。
6. 食べる時間と考える時間を関数 sleep で待つ。

正しく動いたら、今度は左の箸と右の箸を取る間に待ち時間を入れて見る。クライアントの一人の考える時間を0にするのもいい。