

## Cours 8 - Modélisation de la concurrence

### Concurrence:

- Concepts anciens  $\rightarrow$  connu, bien étudié
- Besoins récents - multi-coeur, GPU..
- Difficile :o

### Problèmes:

En *séquentiel*:

- On peut maintenir un lien entre modélisation et implémentation (contrats, MBT, etc)
- On peut maintenir un lien **formel** - logique de Hoare, méthodes formelles (B, Z, cf. SVP en M2)

En *concurrence*:

- Difficile (voire impossible) de maintenir un lien ( $\Rightarrow$  domaine de recherche)
- Comment tester? (interférences entre le test et l'implémentation)
- Comment débbugger?

### Comment modéliser?

#### Formalismes:

- Réseaux de Pétri ( $\sim$  automates concurrents)
- State Charts (UML), tiré des Message Sequencing Charts
- Automates temporisés (UPPAAL)
- Algèbres/Calculs de processus - langages de modélisation
  - CCS, CSP, Pi-calcul (théorique)
  - LOTOS,  $\mu$ -crl2, PROMELA (pratique)

### Outils d'analyse

#### Model-checking:

- Analyse l'espace d'état ( $\sim$  test exhaustif)
- Vérification de propriétés

- sur les états
- sur les suites de transitions (chemins)

**Vérification au runtime:** ( $\sim$  contrats)