

Modelli della Concorrenza

UniShare

Davide Cozzi
@dlcgold

Indice

1	Introduzione	2
2	Introduzione alla concorrenza	3
3	Correttezza di programmi sequenziali	5

Capitolo 1

Introduzione

Questi appunti sono presi a lezione. Per quanto sia stata fatta una revisione è altamente probabile (praticamente certo) che possano contenere errori, sia di stampa che di vero e proprio contenuto. Per eventuali proposte di correzione effettuare una pull request. Link: <https://github.com/dlcgold/Appunti>.

Le immagini presenti in questi appunti sono tratte dalle slides del corso e tutti i diritti delle stesse sono da destinarsi ai docenti del corso stesso.

Capitolo 2

Introduzione alla concorrenza

La **concorrenza** è presente in diversi aspetti della quotidianità (anche non informatica) e un primo esempio di **sistema concorrente** è quella della *cellula vivente*, che può essere vista come un dispositivo che trasforma e manipola dati per ottenere un risultato. I vari processi all'interno di una cellula avvengono in modo concorrente. La cellula è un *sistema asincrono*. Un secondo esempio non informatico è quello dell'*orchestra musicale* dove i vari componenti suonano spesso simultaneamente, rappresentando un *sistema sincrono* (ovvero un sistema che funziona avendo una sorta di “cronometro” condiviso dai vari attori del sistema). Un esempio informatico è un *processore multicores* (anche se in realtà anche se fosse *monocore* sarebbe comunque un sistema concorrente per ovvie ragioni). Anche un *rete di calcolatori* è un modello concorrente. Anche i *modelli sociali umani* sono modelli concorrenti.

I modelli concorrenti hanno alcuni aspetti comuni, semplificando molto:

- competizione per l'accesso a risorse condivise
- cooperazione per un fine comune (che può portare a competizione)
- coordinamento di attività diverse
- sincronia e asincronia

Lo studio e la progettazione di sistemi concorrenti si hanno diversi problemi peculiari che rendono la progettazione degli stessi molto difficile. Un sistema concorrente mal progettato può avere effetti catastrofici.

Per poter sviluppare modelli concorrenti se necessità innanzitutto di:

- **linguaggi**, per specificare sistemi concorrenti. Tra questi si hanno innanzitutto i **linguaggi di programmazione** (con l'uso di *thread*, *mutex*, scambio di messaggi etc... con i vari problemi di

race condition, uso di variabili condivise etc...). Un altro linguaggio (non informatico) può essere quello di una *partitura musicale* (dove si visualizza bene la natura *sincrona*). Un linguaggio per rappresentare un *processo concorrente* è quello di usare un **task graph** (*grafo delle attività*), dove i nodi sono le attività (o eventi) mentre gli archi rappresentano una *relazione d'ordine parziale*, come per esempio una *relazione di precedenza*, sui nodi. Un altro linguaggio è dato dalle **algebre di processi**, simile ad un sistema di equazioni, con simboli che rappresentano eventi del sistema concorrente e operatori atti a comporre fra loro i vari sottoprocessi del sistema concorrente. Ogni “equazione” descrive un processo che costituisce un elemento di un sistema concorrente

- **modelli**, per modellare sistemi concorrenti in astratto. Un esempio è dato dalle **reti di Petri**, che modellano un sistema concorrente partendo dalle nozioni di *stato locale* di uno dei componenti del sistema e di *evento locale* che ha un effetto su alcune componenti (e non tutte). Si ha quindi rappresentato un *sistema dinamico* che si evolve nel tempo (evoluzione rappresentata tramite *relazioni di flusso*)
- **logica**, per analizzare e specificare sistemi concorrenti
- **model-checking**, per validare formule relative a proprietà di sistemi concorrenti

Capitolo 3

Correttezza di programmi sequenziali