Princípy tvorby softvéru
Konkurentnosť and paralelizmus

Robert Lukoťka
lukotka@dcs.fmph.uniba.sk
www.dcs.fmph.uniba.sk/~lukotka

M-255

Konkurentnosť and paralelizmus

- **Konkurentnosť** - schopnosť programu vykonávať kroky rôznych časti programu/algoritmu v ľubovolnom poradí/ na základe čiastocného usporiadania bez zmeny výsledkov/následkov.
- **Paralelizmus** - schopnosť robiť výpočty súčastne.

Konkurentnosť and paralelizmus

- Konkurentnosť je, až na veľmi špeciálne prípady, nevyhnutný predpoklad pre umožnenie paralelizmu
- Konkurentnosť je však užitočná aj bez paralelneho vykonávania (prečo by sme mohli chcieť mať viac threadov aj keď máme k dispozícii iba jeden procesor):
  - Efektivnejšie využitie zdrojov - keď vlákno na niečo čaká, môže sa vykovávať iné vlákno.
  - Konkurentnosť sa dá dosiahnuť aj bez threadov, spomeňte si na systémové volanie select

Race conditions

What could possibly go wrong?

```
int etx_rcvd = FALSE;
void WaitForInterrupt()
{
    etx_rcvd = FALSE;
    while (!ext_rcvd)
    {
        counter++;
    }
}
```

Race conditions

Preložené na

```
int etx_rcvd = FALSE;
void WaitForInterrupt()
{
    while (1)
    {
        counter++;
    }
}
```

Race conditions

OK, toto bola trápna issue. Teraz typickejčie príklady, čo z tohoto sa môže pokaziť (+ si predstavte podobne priklady v Pythone):

```
if (x == 5) // The "Check"
   y = x * 2;

x = x + 1;

x += 1;

x++;
```

Race conditions

- Mutual exclusion (mutex, synchronized, locks, . . .)
- Zabrániť optimalizáciám (volatile, . . .)

Therac 25 - technicky, zdrojom problému bola race condition.

Concurrency konštrukty

- Python - Threading
- Java - BlockingQueue
- ThreadPools
- Event loops

Race conditions

Programovať s Lockami je ťažké

- Race conditions sa netýkajú len pamäte: Shared output devices
- Performance issues
- Race Conditions
- Deadlocks,
- Nechcete aktívne čakať? Spomínate na systémové volanie select?

Concurrent computing

Niektoré good practices:

- Minimal locks (čas aj priestor)
- Prefer higher level constructs.
- Local variables.
- Immutable types.
- Pure functions.
- Získať locky na začiatku v abecednom poradí.
- **Minimalizovať používanie lockovacích konštruktov.**
- **Používať lockovacie konštrukty čo najjednoduchším a najprehľadnejším spôsobom** - konkurentnosť sa fakt blbo testuje.

Príklad

Naozaj minimálne lockovanie môže byť ťažké dosiahnuť:

- Double Checked Locking

Veľa lockov (resp. aj jeden lock na nevhodnom kritickom mieste) však spôsobí, že program sa správa konkeby bežal sekvenčne.

Čo ak potrebujeme lock na dlhšie?

Najmä, ak je scope locku priveľký:

- lock + uloženie stavu, dlhá operácia, lock + kontrola zmeny

Porobne ako prechádzame dlhým DB transakciám.

Architectonické / dizajové riešenia

- Nechať to na databázu
- Vytvoriť single threaded bubliny.
  - BlockingQueue
  - Reactor
- Immutable datové štruktúry - umožňujú bezpečné čítanie a atomické zmeny.
- Assynchronne programovanie / Message passing
- ...

Concurency paterns

Design patterny, pre konkurenciu

- Thread pool (object pool s threadmi)
- Active object
- Reactor
- Double checked locking
- ...

Immutable data types

- Nemodifikovateľné dátové štruktúry.
- Implementácia založená na stromoch s veľkým vetvenim.
- Rôzne hodnoty môžu zdieľať svoju štruktúru.
- Kopírujú sa iba tie časti stromov, kde je to nevyhnutné.

Immutable data types

+ Čítanie je vždy safe.
+ Pripravené pre paralelne algoritmy.
+ Ľahká kontrola, či sa časť stavu zmenila (porovnanie sperníkov).
+ Atomické zmeny stavu.
- Pomalšie
- Potrebujeme garbage collector / alebo počítanie referencií
- Problém so syntaxou.

Asynchronous computing

Ako zavolať funkciu asynchrónne a dozvedieť sa neskôr výsledok

- Futures - môže byť trošku otravné ak nemáme jasný bod, kde budeme čakať na výsledky
- Callbacks
- Promises - cca Futures + .then
  - .then pridáva Promisu callback
  - vracia nový Promise

const promise2 = promise.then(successCallback, failureCallback);

Odporúčané video

Rob Pike - Concurrency Is Not Parallelism

Ďalšie zdroje

- Concurrency - Wikipedia
- Parallel Computing - Wikipedia
- Race Condition - Wikipedia