Concurrency

Synchronization

Pemrograman Berorientasi Objek

Riza Satria Perdana, S.T., M.T.

Teknik Informatika - STEI ITB

Synchronization

- Thread berkomunikasi dengan sharing akses ke field dan object reference fields
- Bisa menimbulkan thread interference dan memory consistency errors
- Thread interference, error yang ditimbulkan ketika sejumlah thread mengakses data bersama (shared)

Synchronization

- Memory consistency errors, error yang diakibatkan view yang tidak konsisten terhadap data bersama (shared)
- Tools yang diperlukan untuk mencegah terjadinya error tersebut adalah sinkronisasi (synchronization)

```
public class Counter {
    private int c = 0;

    public void increment() {
        c++;
    }

    public void decrement() {
        c--;
    }

    public int value() {
        return c;
    }

}
```

Contoh

- Increment:
  - Baca nilai c
  - Tambahkan 1
  - Simpan ke c

Persoalan Thread Interference

- Nilai c=0
- Thread A: Baca nilai c
- Thread B: Baca nilai c
- Thread A: Tambahkan 1
- Thread B: Kurangi 1
- Thread A: Simpan ke c
- Thread B: Simpan ke c

Memory Consistency Errors

- int counter = 0
- Thread A: counter++
- Thread B: System.out.println(counter)

Synchronized Methods

```
public class SynchronizedCounter {
    private int c = 0;

    public synchronized void increment() {
        c++;
    }

    public synchronized void decrement() {
        c--;
    }

    public synchronized int value() {
        return c;
    }
}
```

Synchronized Statements

```
public void addName(String name) {
    synchronized(this) {
        lastName = name;
        nameCount++;
    }
    nameList.add(name);
}
```

Synchronized Statements

```
public class MsLunch {
    private long c1 = 0;
    private long c2 = 0;
    private Object lock1 = new Object();
    private Object lock2 = new Object();

    public void inc1() {
        synchronized(lock1) {
            c1++;
        }
    }

    public void inc2() {
        synchronized(lock2) {
            c2++;
        }
    }
}
```

Liveness

- Kemampuan aplikasi konkuren untuk berjalan dari waktu ke waktu
- Liveness problem: deadlock, starvation, dan livelock
- Deadlock: sebuah situasi dimana dua atau lebih thread terblok selamanya

Liveness

- Starvation: sebuah situasi dimana sebuah thread tidak bisa mendapatkan akses ke resource bersama (shared) dan tidak ada kemajuan
- Livelock: situasi dimana dua atau lebih thread tidak mengalami kemajuan namun tidak terblok

Terima Kasih