

# Multithreading

---

CSSE 221

Fundamentals of Software Development Honors

Rose-Hulman Institute of Technology

# This week: VectorGraphics

---

- Tuesday:
  - Lists and Iterators (capsule)
- Today:
  - Threads (capsule)
  - Project workday
- Friday:
  - Stacks and Queues
  - Sets and Maps

Joe Armstrong, Programming in Erlang

# The World is Concurrent

# Multithreading

- A technique to:
  1. Run multiple pieces of code “simultaneously” on a single machine

Time → Slices	1	2	3	4	5	6	7	8	9	10	11	12	13	14
running thread 1														
running thread 2														

2. Run different parts of a program on different processor cores

# Running Our Own Code Concurrently

From *java.lang*

```
Thread
static void sleep(long ms)
void start()
void interrupt()
...
```

```
«interface»
Runnable
void run()
```

```
MyRunnable
Class
```

```
public class R implements Runnable {
...
public void run() {
    while (true) {
        ... maybe Thread.sleep(...);
    }
}
}
```

Wherever you want to start the Thread:  
`new Thread(object of type R).start();`



Demo

# Animation with Threads

---

- Example 1: A single object
  - “Animate” it with button clicks (done)
  - Animate it with a Timer

```
Timer timer = new Timer(50, animatorButton);  
timer.start();
```

- Animate it by  
using a thread

```
public class R implements Runnable {  
    ...  
    public void run() {  
        while (true) {  
            ... maybe Thread.sleep(...);  
        }  
    }  
}
```

Wherever you want to start the Thread:

```
new Thread(object of type R).start();
```

# Animation with Threads

---

- Example 2: Multiple objects
  - Use separate thread for each object's "brain"
  - Another thread asks Java to update the GUI



<http://www.roadsideamerica.com/story/8543>

# Other thread methods

---

- `.interrupt()`
- `.sleep()`: throws an `InterruptedException` is interrupted
- `.join()`: to wait for a thread to finish before continuing
- `.run()`: Don't call it directly! What would happen if I did?

# Other Uses for Threads

---

- Web servers: many users connecting
  - Any client-server architecture
- Desktop applications:
  - layout, spellchecking, auto-save, ...
- Scientific computing
- Weather forecasting
  - Any simulation (hint, hint)

# Caution!

---

- What if one thread is in the middle of performing an action when its time slice ends?
- What if a second thread's action interferes the first's action?
- Demo bank example