continuing ex from last time

more common format

it (cond 1)  
  stmt;

else it (cond 2)  
  stmt;

else it (cond 3)  
  stmt;

else  
  stmt;
Recall: ternary conditional as:

```
coud ? exp1 : exp2
```

```
Ex
int x, y = 6;

x = (y>5) ? 10 : 20; // x gets 10
```

Equivalent to

```
int x, y = 6;
if (y > 5)
    x = 10;
else
    x = 20;
```

This operator can also be cascaded

```
Ex
int x, y = 6;

x = (y<5) ? 10 : (y<10) ? 20 : (y<15) ? 30 : 40;
// x gets 20
```
Iterative Control Structures

- while
  - to
- do-while

while (cond) {
    stmt;
    stmt;
    stmt;
    stmt;
}

loop repetition condition (LRC)

loop body

Ex. Squares1.java
A very common while loop:

```latex
i = 0; \leftarrow \text{initialize LCV}

\text{while} \ (i < n) \{
  \text{test LRC}
  \quad \text{// do stuff}
  \quad i++ \leftarrow \text{increment LCV}
\}
```

Equivalent for loop:

```latex
\text{for} \ (i = 0; \ i < n; \ i++) \{
  \quad \text{// do stuff} \quad \text{loop body}
\}
```

Ex. Square32.java

general form of do-while:

do {
  stmt;
  stmt;
  ... 
  stmt;
} while (cond);

loop body

LRC

Note: While loop body may execute 0 times, but do-while loop body must execute at least once.

Ex. Square33.java
Logic errors:

```java
i = 0;
while (i < 10); // actually an infinite loop
    // stuff
    i++;
}
```

de (i = 0; i < 10; i++);
    // stuff
}
```

Ex. Average.java