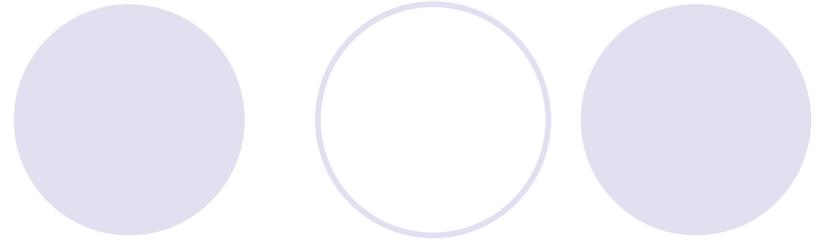


Boolean Expressions and Conditional Execution

Chapter 5

For Next Time

- Read Chapter 5



Relational Operators

- Compare two expressions
- Evaluate to Boolean values: `true`, `false`

Operator	Meaning
<code>==</code>	is equal to
<code>>=</code>	is greater than or equal to
<code><=</code>	is less than or equal to
<code><</code>	is less than
<code>></code>	is greater than
<code>!=</code>	is not equal to

Notes



- Test #1 is ***two weeks*** from today
- Begin Lab 4 ***immediately*** after finishing Lab 3
- Visit the tutors!

Boolean Operators

- Combine simpler Boolean expressions to make compound Boolean expressions

e_1	e_2	$e_1 \ \&\& \ e_2$ (and)	$e_1 \ \ e_2$ (or)	$! \ e_1$ (not)
false	false	false	false	true
false	true	false	true	true
true	false	false	true	false
true	true	true	true	false

Evaluate the Following Expressions

```
int x = 5, y = 2;
```

```
x >= 5
```

```
x > 5
```

```
x == 5
```

```
x == y
```

```
x != y
```

```
x >= 5 && y < 10
```

```
x >= 5 || y < 10
```

```
x < 0 && y > 10
```

```
x < 0 || y > 10
```

```
x > 0 && y > 10
```

```
x > 0 || y > 10
```

```
x > 0 || y < 10
```

```
x > 0 && x < 10
```

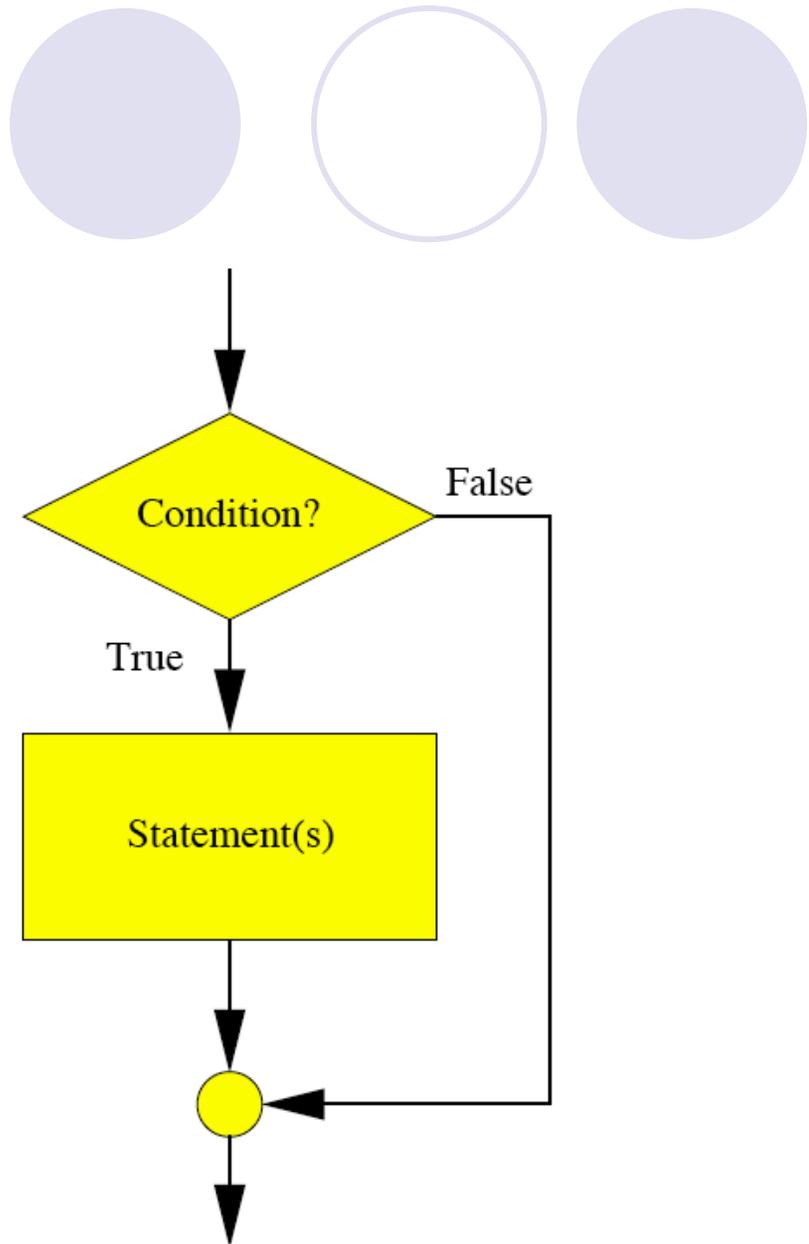
```
x > 0 || x < 10
```

```
x < 0 && x > 10
```

if Statement

- Conditionally execute sections of code based on the value of a Boolean condition

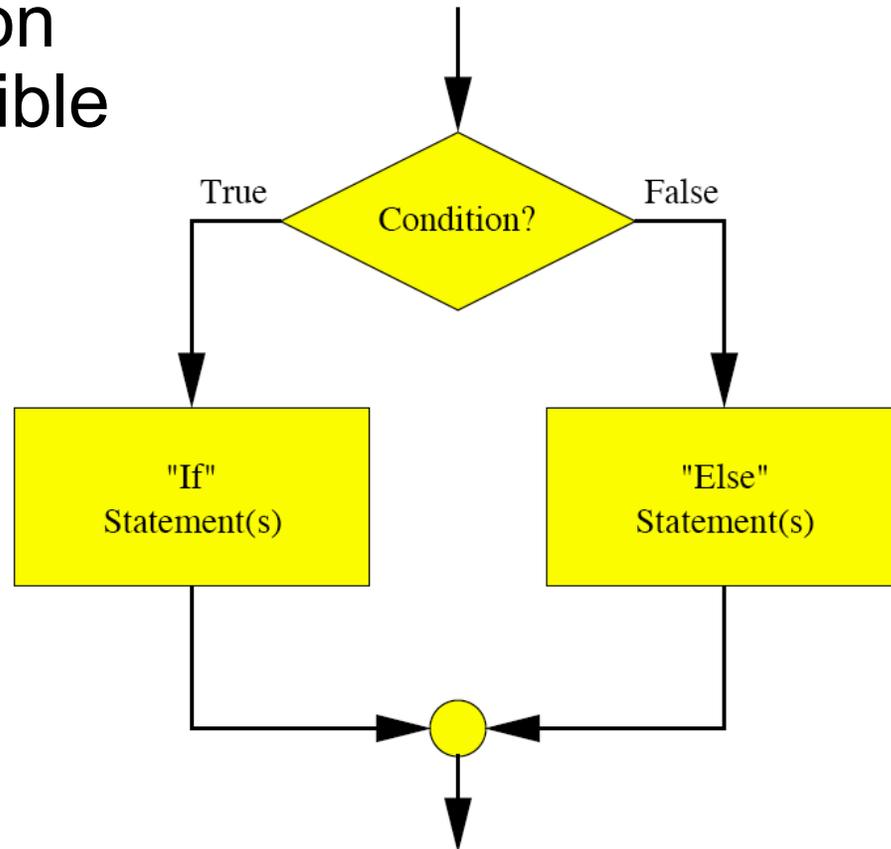
```
if ( condition )  
    body
```



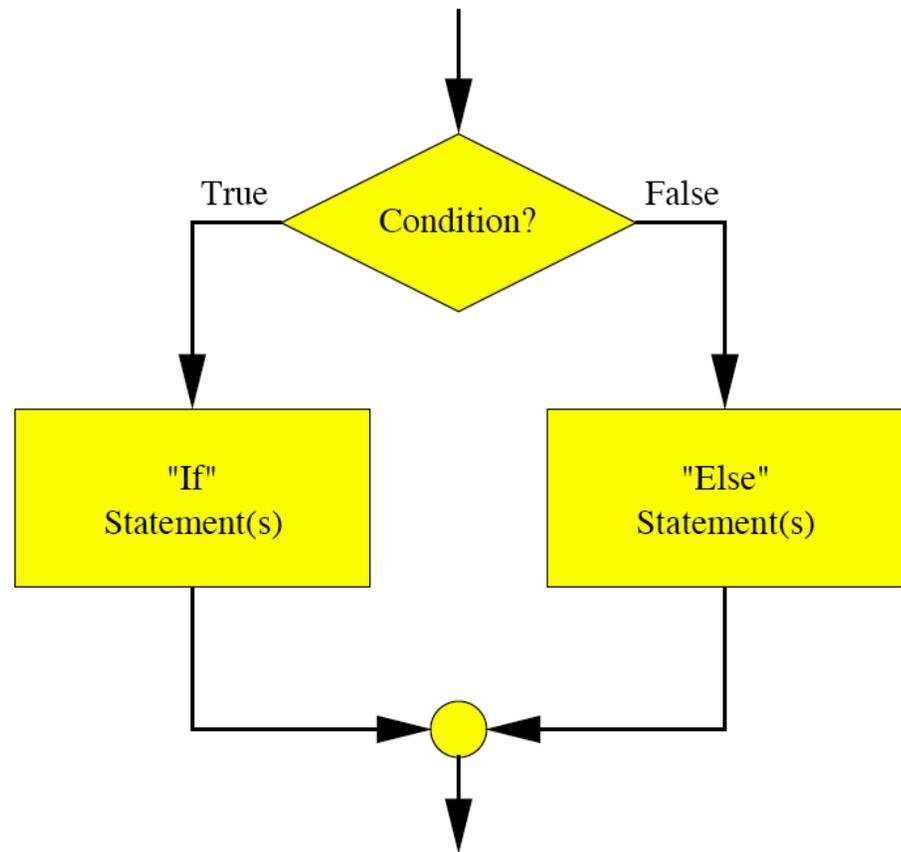
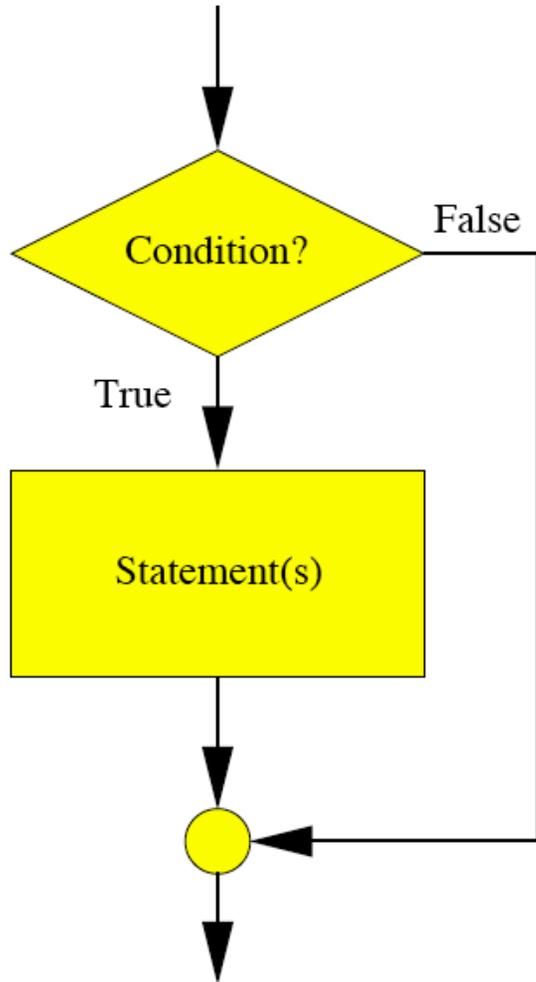
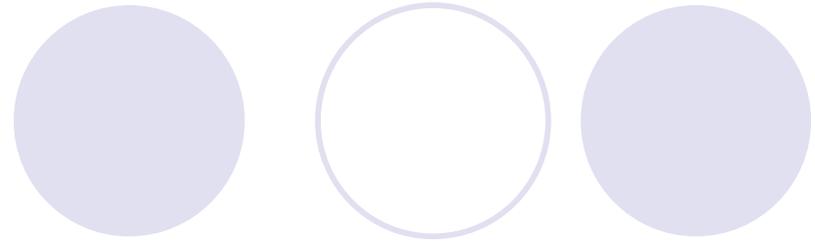
if/else Statement

- Select one execution path from two possible execution paths

```
if ( condition )  
    ifbody  
else  
    elsebody
```



if vs. if/else





More complex conditionals