



TUTORIAL 4

BME 121 2016

Jeff Luo


Topics

- Inline If Statement

Computer Decision Making So Far

```
if ( // test1 )
{
    // test1 == true
}
else if ( // test2 )
{
    // test1 == false && test2 == true
}
else
{
    // otherwise
}
```

*Final value must be a string or int



```
switch ( // expression )
{
    case value1:
        // expression == value1
        break;
    case value2:
        // expression == value2
        break;
    default:
        // otherwise
        break;
}
```


Inline If Statement

- A sometimes-useful substitution for an if statement block
- The entire inline if statement **evaluates to a single value**
- Three parts:
 - A **test**, followed by a **question mark**
 - followed by the **value** that the whole statement would be evaluated to if test == true, followed by a **colon**
 - Followed by the **value** that the whole statement would be evaluated to if test == false
- Basic format:

```
// test ? // true value : // false value
```

- Example:

```
int x = 5 > 10 ? 15 : 30 ;
```



evaluates to 30

- Equivalent if statement block:

```
int x;  
if ( 5 > 10 )  
{  
    x = 15;  
}  
else  
{  
    x = 30;  
}
```



Examples:

```
int MyMax(int x, int y)
{
    return x > y ? x : y;
}
```

```
int MyMin(int x, int y)
{
    return x < y ? x : y;
}
```

```
string Winner(int p1, int p2)
{
    return p1 >= p2 ? "Player 1" : "Player 2";
}
```

Nested Inline If Statement

```
string SmallestMinWinner(int p1Min, int p2Min)
{
    return p1Min < p2Min ? "Player 1" :
        (p1Min == p2Min ? "Tie" : "Player 2");
}
```

- Read:

- First check if $p1Min < p2Min$, if so return "Player 1"
- If not, check to see if they are equal
- If they are equal, return "Tie"
- If they are not equal (ie $p1Min > p2Min$) return "Player 2"