



# CS 161

## Intro to CS I

Finish Conditionals/Begin Loops



# Odds and Ends

- Last week to demo Assignment 1
- Demo Assignment 2, extra slot
- Peer Reviews due Thursday night
- Assignment 3 posted, design due Sunday
  
- Questions?

# Reflection



Oregon State University  
College of Engineering

- Tell me what this program does...

```
1 #include <iostream>
2 #include <climits>
3
4 using namespace std;
5
6 int main() {
7     int input;
8
9     cout << "Enter an integer: ";
10    cin >> input;
11
12    //produces a description about the number entered by the user
13    if(input%2)
14        cout << "odd integer, ";
15        if(input<(INT_MAX/2))
16            cout << "less than half the largest an int can be!" << endl;
17        else
18            cout << "less than half the largest an int can be!" << endl;
19    else
20        cout << "even integer, ";
21        if(input<(INT_MAX/2))
22            cout << "less than half the largest an int can be!" << endl;
23        else
24            cout << "greater than half the largest an int can be!" << endl;
25
26    return 0;
27 }
```



# Logical Operators

- AND
- OR
- NOT

Are all logical operators binary?

What is short circuiting?

When might you use it?

# Demo...



**Oregon State University**  
College of Engineering



# We can use a switch...

```
switch( <expression> ) {  
    case <const-expr> :  
        <statement>;  
    ...  
    case <const-expr> :  
        <statement>;  
    ...  
    default:  
        <statement>;  
    ...  
}
```



# C++ Switch Example

```
switch( x ) {  
    case 0:  
        std::cout <<"X is zero";  
        break;  
    case 1:  
        std::cout <<"X is one";  
        break;  
    case 2:  
        std::cout <<"X is two";  
        break;  
    default:  
        std::cout <<"You have entered an invalid number!";  
}
```





# C++ Switch Example

```
switch( x ) {  
    case 0:  
    case 1:  
        std::cout << "X is zero or one";  
        break;  
    case 2:  
        std::cout << "X is two";  
        break;  
    default:  
        std::cout << "You have entered an invalid number!";  
}
```

# Multiple Decisions



- What if I want to make these same decisions for the whole year or while we can still ski?

If it is sunny today

then I'll go to the beach

if it is windy at the beach

then I'll fly a kite

else if it is not windy at the beach

then I'll walk on the shore

Else if it is raining today

then I'll stay inside and read a book

Else if it is snowing

then I'll go to the mountains to ski

- Repeat the process for 365 days
- Repeat the process while I can still ski 😊



# How do we do this for a year?

- Repetition: for loops
  - Semantics
    - Repeat for a specific # of iterations w/ starting point, ending point, and a way to get from start to end
  - Syntax

```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

# How do we do this while we can still ski?



- Repetition: while loops
  - Semantics
    - Repeat while something continues to hold true
  - Syntax

```
bool can_ski=true;
while(can_ski == true) {
    //go skiing
    cout << "Can you still ski? (0-false, 1-true)" << endl;
    cin >> can_ski;
}
```



# The for Loop Pattern

```
for(<variable> = n; <variable> <= p; <variable>++)  
{  
    <statement>;  
    ...  
}
```

```
for(<variable> = n; <variable> >= p; <variable>--){  
    <statement>;  
    ...  
}
```



# The for Loop

Starting point:  
Initialization

```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

# The for Loop



Ending point:  
Continuation Test

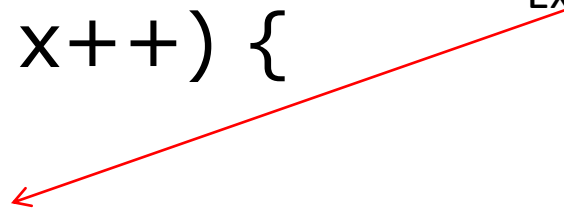
```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```



# The for Loop

```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

Test is True:  
Execution Block



- What do you notice about order?





# The for Loop

Increment:  
Update

```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

- Same as  $x = x + 1$
- What about  $x = x + 2$ ?



# The for Loop

Ending point:  
Continuation Test

```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

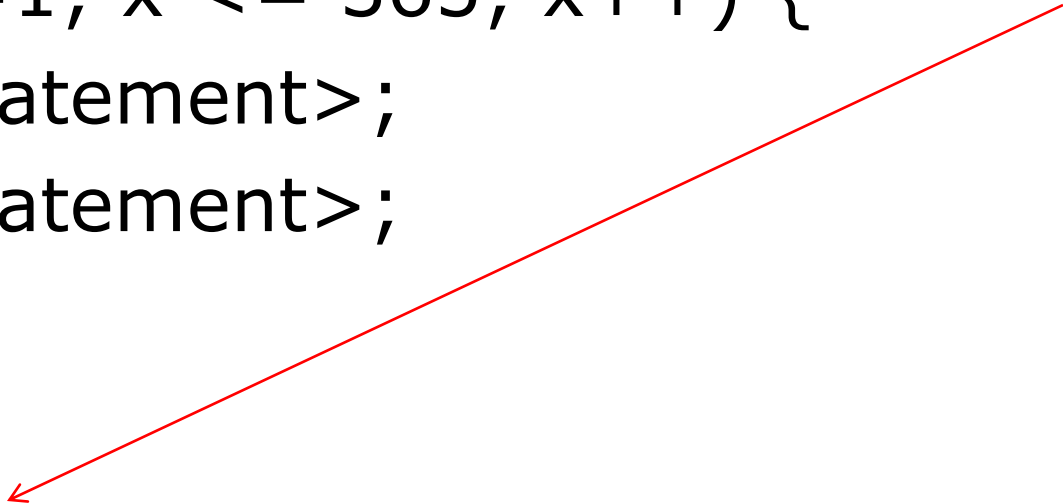
- What do you notice about order?

# The for Loop



```
for(x=1; x <= 365; x++) {  
    <statement>;  
    <statement>;  
    ...  
}
```

Test is False:  
Execution  
after loop





# The for Loop Examples

```
for(x=0; x <= 100; x++)  
    cout << "hello world\n";  
for(x=0; x < 100; x++)  
    cout << "hello world\n";  
for(x=-100; x <= -1; x++)  
    cout << "hello world\n";  
for(x=-100; x <= 100; x++)  
    cout << "hello world\n";
```



# The for Loop Examples

```
for(x=1; x <= 1; x++) {  
    cout << "hello world\n";  
}
```

```
for(x=1; x < 1; x++) {  
    cout << "hello world\n";  
}
```

- Why is it better to use curly braces?



# The while loop

```
for(x=1; x <= 100; x++)  
    cout << "hello world\n";
```

**VS.**

```
int x=1;  
while(x<=100) {  
    cout << "hello world\n";  
    x++;  
}
```



# Common Mistakes

```
int x=1; ← What if we forget this?  
while(x<=100) {  
    cout << "hello world\n";  
    x++; ← What if we forget this?  
}
```



# The do/while loop

```
int x=1;
do {
    cout << "hello world\n";
    x++;
} while(x<=100);
```

- Difference b/w while and do/while?





# Nested for Loops

```
for(x = 0; x < 10; x++) {  
    for(y = 0; y < 10; y++) {  
        cout << "hello world\n";  
    }  
}
```

- How many times is Hello World printed?



# Reuse Variables

```
for(x = 0; x < 10; x++) {  
    cout << "The value of x is: " << x << endl;  
}
```

```
for(x = 0; x < 10; x++) {  
    cout << "The value of x is: " << x << endl;  
}
```



# Variables with same name

```
int x;  
for(x = 0; x < 10; x++) {  
    for(x = 0; x < 10; x++) {  
        cout << "The value of x is: " << x << endl;  
    }  
}
```

- What is the output from this nested loop?



# Infinite Loops

```
int x;  
for(x = 0; x < 10; x++) {  
    for(x = 0; x < 5; x++) {  
        cout << "The value of x is: " << x << endl;  
    }  
}
```



# Infinite Loops

```
int x, y;  
for(x = 0; x < 10; x++) {  
    for(y = 0; y < 5; x++) {  
        cout << "The value of x is: " << x << endl;  
    }  
}
```

# Infinite Loops



```
int x, y;  
for(x = 0; x < 10; x++) {  
    for(y = 0; x < 5; y++) {  
        cout << "The value of x is: " << x << endl;  
    }  
}
```



# Infinite Loops

```
int x;  
for(x = 1; x <= 10; x++) {  
    cout << "The value of x is: " << --x << endl;  
}
```