

Slides #5.1

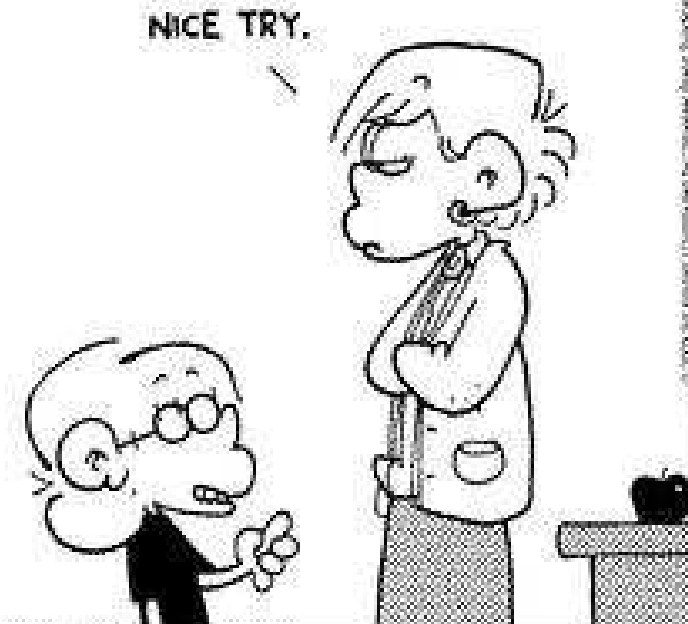
# If Statements

Chapter 2.4 – 3.4 (some parts)

```
#include <stdio.h>
int main(void)
{
    int count;

    for (count = 1; count <= 500; count++)
        printf("I will not throw paper airplanes in class.");

    return 0;
}
```



# Topics

- 1) How can we work with **true** and **false**?
- 2) How can write **if statements**?
- 3) **Loops**: **while** and **for**

# Boolean Expressions

# Boolean Expressions

- Boolean Expressions evaluate to...
  - Called a “condition.”
- Most often used with **if** or loop (**while**, **for**, **do..**)

```
int loops = 45;
// Check loop length
if (loops > 30) {
    cout << "This may take a while!\n";
}
// Count down
while (loops != 0) {
    cout << "Counting down... " << loops << endl;
    loops = loops - 1;
}
```

# Truth?

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Dear C++, what is the truth?\n";
    cout << "  Is 2 greater than 5?  = " << (2 > 5) << endl;
    cout << "  Is 2 less than 5?     = " << (2 < 5) << endl;
    return (0);
}
```

- How C++ interprets values:
  - What is false:
    -
  - What is true:
    -

## Aside:

You can also display “true” and “false” by first using:  
`cout << boolalpha;`

```
Dear C++, what is the truth?
  Is 2 greater than 5? =
  Is 2 less than 5?   =
```

# Equality

- Equality Operators:

- Equal:

- Not Equal:

- Examples:

```
cout << (0 == 0);    // 1
```

```
cout << (65 != 65);  // 0
```

```
cout << (42 == 2);   // 0
```

```
cout << ('b' != 'a'); // 1
```

- Equality vs Assignment:

- if the values are the same (**==**).

- the values the same (**=**).

# Relational Operators

- Compare two values to see which is less/greater:
  - < Less than
  - <= Less than or equal to
  - > Greater than
  - >= Greater than or equal to
- Examples:
  - cout << (age >= 65); // 1 if senior
  - if (age >= 65) {...} // Check for senior
  - if (value < 0) {...} //..
  - if (0 > value) {...} // Also check for neg. value
  - if (value <= 0) {...} // Check for non-positive value





# If Statement

- The `if` statement performs conditional execution once:

```
cout << "Value is negative.\n";
```

- **Style**
  - Indent the “then” statements to make it easy to read. (Compiler does not care!)
  - Good style...

# if, if-else

- ..  
if (age < 18) {  
 cout << "Minor";  
}

- ..  
if (age < 18) {  
 cout << "Minor";  
} else {  
 cout << "Adult";  
}

- ..  
if (age < 18) {  
 cout << "Minor";  
} else if (age < 65) {  
 cout << "Adult";  
} else {  
 cout << "Senior";  
}

# Nested if

if statements can be..

other if statements

```
int main() {
    cout << "Enter your shoe size: ";
    int shoeSize = 0;
    cin >> shoeSize;

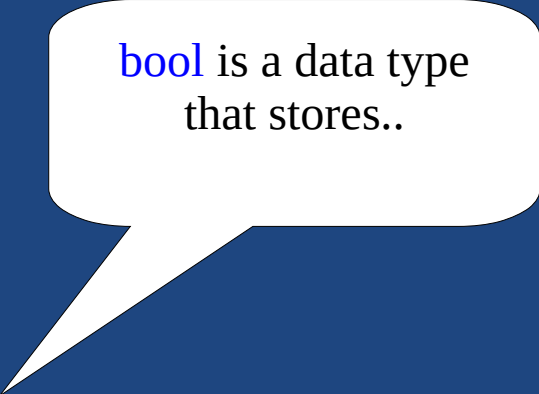
    // Print summary
    if (shoeSize > 0) {
        if (shoeSize > 10) {
            cout << "Big feet!\n";
        } else {
            cout << "Small feet!\n";
        }
    } else {
        cout << "Invalid size!\n";
    }
}
```

```
// Can code without nested if:
if (shoeSize <= 0) {
    cout << "Invalid\n";
} else if (shoeSize > 10) {
    cout << "Big\n";
} else {
    cout << "Small!\n";
}
```

# Multiple statements

- Use `{...}` to place multiple statements in the `if` or `else`:

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Enter your IQ: ";
    int iqValue;
    cin >> iqValue;
    bool genius = false;
    if (iqValue > 150)
        genius = true;
    else
        genius = false;
    cout << "Not so smart\n";
}
```



`bool` is a data type  
that stores..

- This code will...
- Must use block statement `{...}` with multiple statements.

# Better code (aside)

- Note there is better way to write the logic:

## Previous Solution

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Enter your IQ: ";
    int iqValue;
    cin >> iqValue;
    bool genius = false;
    if (iqValue > 150)
        genius = true;
    else {
        genius = false;
        cout << "Not so smart\n";
    }
}
```

## Better Solution

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Enter your IQ: ";
    int iqValue = 0;
    cin >> iqValue;

    bool genius = (iqValue > 150);
    if (!genius) {
        cout << "Not so smart\n";
    }
}
```

No if  
required!

! means not

# Common Errors

- 

```
if (a < b); {  
    cout << "less\n";  
}
```

Extra ';' ends the `if` statement  
(creates a.. ..).  
`cout` will always execute.

- 

```
if (a = b) {  
    cout << "Equal!\n";  
}
```

`a = b` is the..  
Should be `a == b`.  
VERY common bug!

# Scope

- Variables exist in a scope:
  - Variables defined in a block...

- What are the errors?

```
if (a > b) {  
    int fromThen = 1;  
} else {  
    cout << fromThen;  
    int fromElse = 2;  
}  
cout << fromThen;  
cout << fromElse;
```

# Exercise

- Complete this program to tell the user if their number is negative, positive, or zero.
- 3 Sample outputs:

```
What's the number? 42  
Positive
```

```
What's the number? 0  
Zero
```

```
What's the number? -3  
Negative
```

```
int main()  
{  
    double favNum = 0;  
    cout << "What's the number? ";  
    cin >> favNum;  
  
    // Your code here!  
  
}
```



# Summary

- Boolean expressions evaluate to **true** or **false**.
  - Comparison: **==**, **!=**, **>**, **<**, **>=**, **<=**
- **if-else** statement control program **branching**.
  - Can pick which code to run.