

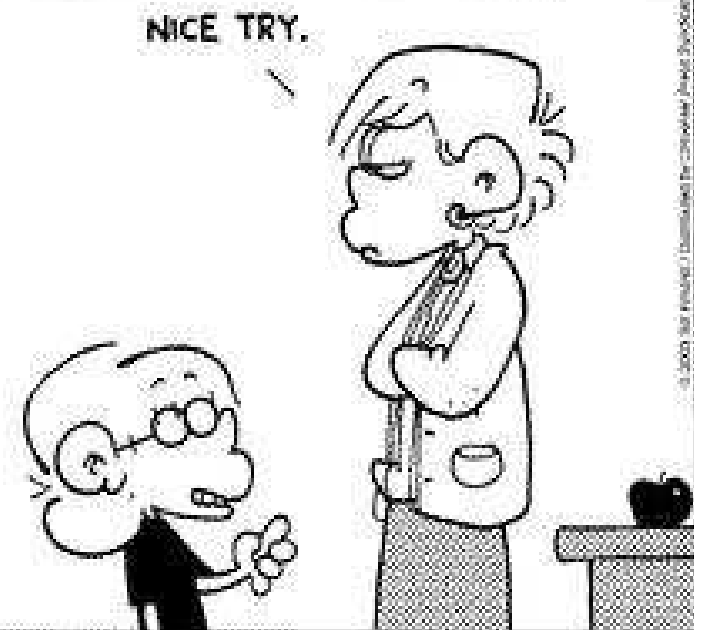
Slides #5.1

# 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**?

# Boolean Expressions

# Boolean Expressions

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

```
// Count down, showing `if` and `while` statements
#include <print>
using namespace std;

int main() {
    const int LOOPS = 45;

    // Check loop length
    if (LOOPS > 30) {
        println("Looping {} times is a lot!", LOOPS);
    }

    // Count down
    int i = LOOPS;
    while (i > 0) {
        println("Counting down... {}", i);
        i--;
    }
}
```

```
Looping 45 times is a lot!
Counting down... 45
Counting down... 44
Counting down... 43
Counting down... 42
Counting down... 41
Counting down... 40
Counting down... 39
Counting down... 38

...
Counting down... 4
Counting down... 3
Counting down... 2
Counting down... 1
```

# Equality

- Equality Operators:

- Equal:
- Not Equal:

- Examples:

```
println("{} ", ( 0 == 0 ));      // true
println("{} ", (65 != 65));     // false
println("{} ", (42 == 2));      // false
println("{} ", ('b' != 'B'));   // true
```

- 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:
  - 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

# Truth?

```
int main() {  
    println("Dear C++, what is the truth?");  
    println(" 2 == 5 = {}", (2 == 5));  
    println(" 2 != 5 = {}", (2 != 5));  
    println(" 2 > 5 = {}", (2 > 5));  
    println(" 2 <= 5 = {}", (2 <= 5));  
  
    if (0) {  
        println(" 0 is true!"); // Not executed  
    }  
    if (1) {  
        println(" 1 is true!");  
    }  
    if (42) {  
        println(" 42 is true!");  
    }  
}
```

```
Dear C++, what is the truth?  
2 == 5 = false  
2 != 5 = true  
2 > 5 = false  
2 <= 5 = true
```

```
1 is true!  
42 is true!
```

- What is false:

- 

- What is true:

-



# If Statement

- The **if** statement performs conditional execution once:

```
int height = 0;  
print("Enter height: ");  
cin >> height;  
  
    println("Height is negative.");
```

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

# if, if-else

If

If-else

If,  
else-if,  
else

```
// Demonstrate if statements
#include <iostream>
#include <print>
using namespace std;

int main() {
    int age = 0;
    print("Enter your age: ");
    cin >> age;

    if (age < 18) {
        println("You are a minor.");
    }

    if (age < 18) {
        println("You are a minor.");
    } else {
        println("You are an adult.");
    }

    if (age < 18) {
        println("You are a minor.");
    } else if (age < 65) {
        println("You are an adult.");
    } else {
        println("You are a senior.");
    }
}
```

Enter your age: 87  
You are an adult.  
You are a senior.

# Nested if

if statements can be..

```
// Comment on the user's shoe size
#include <iostream>
#include <print>
using namespace std;

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

    // Print summary with nested if
    if (shoeSize > 0) {
        if (shoeSize > 10) {
            println("Big feet!");
        } else {
            println("Small feet!");
        }
    } else {
        println("Invalid size!");
    }
}
```

other if statements

```
// Can use if, else-if, else instead
if (shoeSize < 0) {
    println("Invalid size!");
} else if (shoeSize > 10) {
    println("Big feet!");
} else {
    println("Small feet!");
}
```

# Multiple statements

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

```
int main() {  
    // Get user's IQ  
    int iqValue = 0;  
    print("Enter your IQ: ");  
    cin >> iqValue;  
  
    // Decide if a genius  
    bool genius = false;  
    if (iqValue > 150)  
        genius = true;  
    else  
        genius = false;  
    println("Not that smart");  
}
```

**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

```
int main() {  
    // Get user's IQ  
    int iqValue = 0;  
    print("Enter your IQ: ");  
    cin >> iqValue;  
  
    // Decide if a genius  
    bool genius = false;  
    if (iqValue > 150) {  
        genius = true;  
    } else {  
        genius = false;  
        println("Not that smart");  
    }  
}
```

## Better Solution

```
int main() {  
    // Get user's IQ  
    int iqValue = 0;  
    print("Enter your IQ: ");  
    cin >> iqValue;  
  
    // Decide if a genius  
    bool genius = (iqValue > 150);  
  
    ! means not  
    No if required!  
    if (!genius) {  
        println("Not that smart");  
    }  
}
```

# Common Errors

- 

```
if (a < b); {  
    println("a is less than b");  
}
```

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

- 

```
if (a = b) {  
    println("a is equal to b");  
}
```

`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?

```
int main() {  
    if (a > b) {  
        int fromThen = 1;  
    } else {  
        println("{} ", fromElse);  
        int fromElse = 2;  
    }  
    println("{} ", fromThen);  
    println("{} ", fromElse);  
}
```

Error: Must declare variable first.

Error: Variables must be declared in same block.

# 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;  
    print("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.