C++ Data Types

- int - 4 bytes, holds integers
- double - 8 bytes, holds decimal numbers
- char - 1 byte, holds a character
- bool - 1 byte, holds "true" or "false"

```cpp
bool sorting;
char letter;
double pi;
int n;
```

```cpp
sorting = true;
letter = 'a';
pi = 3.14159;
n = 3;
```
C++ Operations

- Addition (+)
  \[ \text{sum} = a + b; \]

- Subtraction (-)
  \[ \text{diff} = a - b; \]

- Multiplication (*)
  \[ \text{prod} = a \times b; \]

- Division (/)
  \[ \text{quot} = a \div b; \]

- Modular Division (%)
  \[ \text{remain} = a \% b; \]

- Subtraction (-)
  \[ \text{diff} = a - b; \]

- Addition (+)
  \[ \text{sum} = a + b; \]
Order of Operations

- and / are evaluated before + and -
- spacing doesn’t matter
- best to use parentheses to make things clear
- and / are evaluated before + and -
- spacing doesn’t matter

\[ m = 18 \]
\[ n = 10 \text{ and} \]
\[ 3 \times (2 + 4) = m \]
\[ 4 + (3 \times 2) = n \]
\[ m = 18 \text{ gives} \]
\[ n = 14 \text{ gives} \]
\[ 3 \times 4 = 2 + 4 \text{ gives} \]
\[ n = 10 \text{ gives} \]
\[ n = 2 \times 3 + 4 \]
Input and Output

- Use `cout` to output text
  ```cpp
cout << "Hello!" << endl;
```
- Use `cin` to input text
  ```cpp
cint a; cin >> a;
```
- Must use `using namespace std;` at the top of your program to use `cout` and `cin`
Use comments to document your program.

Comments

// Single line comment.

/* A multiple line comment should look something like this. */

/* Like this. */
Indenting Style

- Use tabs to indent your code. This helps you to highlight loops and logical constructions of your program.
- Emacs will automatically indent your code for you if you use the tab key to begin a new line of code.
- Use tabs to indent your code. This helps you to highlight loops and logical constructions of your program.
```cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    int n = 1;
    for (i = 1; i <= 5; i++)
    {
        n = n * i;
        cout << i << " n is " << n << endl;
    }
    cout << "Final n value is " << n << endl;
    cout << "Final i value is " << i << endl;
}
```
Final $i$ value is 6
Final $n$ value is 120
$n$ is 120
$i$ is 5
$n$ is 24
$i$ is 4
$n$ is 6
$i$ is 3
$n$ is 2
$i$ is 2
$n$ is 1
$i$ is 1
Boolean Expressions

- There are a variety of comparison operators: `==`, `<`, `<=`, `>`, `>=`, `!=`.
- You can also combine boolean expressions with the AND operator `&&`, the OR operator `||`, and the negation operator `!`.

Boolean Expressions
Boolean Examples

( (\zeta = i \bar{\lambda}) \land (\varepsilon = x) ) \mid \mid (\bar{\lambda} = x) .
(\varepsilon < a) \mid \mid (\varepsilon \Rightarrow a) .
(\varepsilon < \nu) \land (\bar{\tau} = i \nu) .
(\bar{\lambda} > x) \mid \mid (\bar{\lambda} < x) .
(\bar{\lambda} = x) .
if-then-else statements

• if (boolean expression)
  
  {  
    cout << “expression is true” << endl;  
    // other statements can go here  
  }  

else
  
  {  
    cout << “expression is false” << endl;  
    // other statements  
  }
Homework

• Read Chapter 1 in Carrano and Prichard
• Finish Exercises 1 and 2 (website)
• Work on Exercise 3 (website)
• Recall: Website is
  http://albertr.washcoll.edu/~mmclendon2/MAT102/
• Finish Exercises 1 and 2 (website)
• Read Chapter 1 in Carrano and Prichard