C++ Programming Basics
Continued

C++ Lecture 2
Nick Matthews
Adam Kohl
Lecture Goals

- Today is devoted to more basic building blocks
  - Variables
  - User Input
  - Operators
  - Arrays
What Are Variables?

- Variables are boxes for information

- Different types of variables store different types of information
  - bool – a variable type that is either true or false
  - char – a variable type that can hold a single character (e.g., ‘x’, ‘y’, ‘z’)
  - int – a variable type that can hold an integer value (e.g., 1, 2, -3)
  - float - a variable type that can hold a decimal value up to around 7 digits
  - double – a variable type that can hold a decimal value up to 15 digits
  - strings – a variable type that can hold text (requires #include <string>
Variable Syntax

Declaring a variable \(\rightarrow\) [data type] [name]; (Make an empty box)

```cpp
// declare int variable
int a;
```

Initialize the variable \(\rightarrow\) [name] = [value]; (Fill the empty box)

```cpp
// initialize int variable
a = 5;
```

Declare and initialize a variable \(\rightarrow\) [data type] [name] = [value]; (Make and fill a box)

```cpp
// declare AND initialize int variable
int b = 10;
```
Once you create a variable, you can use it later in your program!

```cpp
int main() {
    int a = 10;

    // print variable
    std::cout << a << std::endl;
}
```
Variable Examples

- Declare some variables of different data-types
- Initialize the variables
- Declare and initialize some more variables
- Print them out!
User Command Line Input

- What if we want user input in our program?

- We use the `cin` object to store user input into a variable

```cpp
// cin
int a;
cout << "Enter a number: "; // prompt user
cin >> a; // get user input and store in a
```

- Program will **pause** at `cin` and wait for user input
Mini Task

1. Make a program which asks the user to input a number
2. Store the number in a variable using `cin`
3. Print the number back to the user using `cout`
Operators

๏ What if we want to add or multiply values/variables together?

๏ Arithmetic operators (used on numeric variables)
  - +, -, *, /, ....

๏ Comparison operators (used on boolean variables)
  - ==, !=, <, <=, >=

๏ Logical operators (used on boolean variables)
  - &&, ||, !
Operator Examples

- Using operators during variable assignment
  - int a = (some expression using operators)

- Conditional and Logical operator expressions
  - <, >, <=, >=, ==, &&, ||
Arrays

- Arrays allow us to store multiple variables of the same data-type in a list fashion.

```cpp
int myArray[4]; // Want an array with 4 slots to fill with ints
myArray[0] = 2; // Indexing starts at 0!
myArray[1] = 5; // Assign the second array slot to 5
myArray[2] = 9;
myArray[3] = 3; // This is the 4th slot at index 3

std::cout << "The 3rd value in myArray is: " << myArray[2] << std::endl;

The 3rd value in myArray is: 9
Press any key to continue . . .
```
Array Examples

- Declare an array
- Instantiate each slot of the array individually
- Declare and instantiate an array
- Set a new value in the array
- Access a value in the array and print it out
Questions?
Assignment


2. Under challenges/ read timeConversion.md 📖

3. Make a new project and code 👨💻