Philip Pesic

Week 2

August 28 2022

Week 2 Lecture 2 Notes

IDEs

IDE, or integrated development environment, is an application that contains a collection of tools to help software developers. IDEs can vary in selection of tools, compilers, and compatible languages.

Eg: The Xcode IDE supports C++, and has tools such as a compiler, a debugging tool, and a text editor.

Visual Studio Startup and Use

When opening VS Code, you can choose to create a new project, or open an existing one. Assuming you create a new project, you then must choose your language, console application, project name, and file location. From here, you can begin writing your code. Once your code is written, you can press f5 to debug and run it. If your code runs, it was successful. If error messages pop up, there are bugs in your code. The debug pane is the best place to analyze your problems and rewrite your code.

Xcode Use

Xcode starts similarly to VS Code. Upon opening, you have the same options to select your project, language, etc. The biggest differences come from the GUI. Xcode has its panes laid out differently, and some error messages and similar features may appear differently.

Philip Pesic

Week 2

August 28 2022

Week 2 Lecture 2 Notes

Language Words and Grammar

Programming language has words and grammar, like spoken language. Words are called Key Words, and grammar is the language syntax.

Ex. int age = 15

Libraries

Libraries are collections of pre-written code that programmers can reuse in their code. It reduces complexity of the program and reduces the time programmers spend writing their programs.

Ex. #include <cmath> is a library with common mathematical functions, like sin, cos, exp…

Iostream

Iostream is a library of functions that input data from the keyboard and output data to the terminal.

Ex. cout, cin, endl

Variables (Reserving RAM)

A variable is an addressed place in RAM where a value is stored. Declaring a variable reserves an appropriate amount of memory depending on the data type.

Ex. int age; //integer variable ‘age’ will have a unique address in RAM and will occupy 4 bytes.

Philip Pesic

Week 2

August 28 2022

Week 2 Lecture 2 Notes

Storing (Assigning) values in to RAM memory variables

When a value is assigned to a variable, that value is stored at the address in memory dedicated to that variable.

Ex. age = 15; // value 15 will be stored at the address assigned to variable age

Math Operations

In programming, we can use math operations on variables, such as addition, subtraction, division, multiplication, exponents, etc.

Ex. int a = b * c + d / e

The problem/solution

Declaring a large number of variables is long and inefficient. Arrays help to cut down the time and simplify a large number of variables. They create a parent variable that stores values in an index. Instead of searching for a specific variable, you can search for the index number in the array.

Ex: int array[5] = {0};

Philip Pesic

Week 2

August 28 2022

Week 2 Lecture 2 Notes

Initialization

There are 4 ways to initialize a variable:

Short term: Quickly assigns every variable the same value

Long term: Manually assigns each variable a specific value

Individual: Manually assigns each variable a specific value, but with a longer, yet clearer syntax

Loop: Utilizes a loop to progressively initialize each value

Calculation

Arrays can be used in calculations by using equations to call an index. These equations can be almost anything, as long as they maintain the proper syntax, and stay within the range of the array.

Ex: array[i]; i++;

Multidimensional Arrays

If an array is just a straight line of variables, then two and three dimensional variables can be visualized as a table and cube respectively. Each uses a combination of [] to hold different indexes, and are often used in equations.

Ex: array[2][3] = {{0,1}{0,1,2}};

Philip Pesic

Week 2

August 28 2022

Week 2 Lecture 2 Notes

Declaration Hierarchy

Source code → Compiled into machine level code → Memory manager assigns every byte an address → RAM stores every byte

Ex (Not real machine code or addresses):

cout << “hi”; → 11101010100110010010101101100001 → Byte 1 = Address A, Byte 2 = Address B, Etc. → Byte 1 stored in RAM A, Byte 2 stored in RAM B, Etc.