Comp Sci 302

Jim Williams, PhD
CS 302 Description

Instruction and experience in the use of an object-oriented programming language. Program design; development of good programming style; preparation for other Computer Science courses.

Programming literacy - learn to **read** and **write** Java programs.
First Java Program

```java
public class HelloWorld {

    public static void main(String[] args) {
        System.out.println("Hello World");
    }
}
```
Why Study Computing?

10. Future opportunities in computing are without boundaries.
9. Computing is an essential part of well-rounded academic preparation.
8. Computing has space for both collaborative work and individual effort.
7. Computing offers great opportunities for true creativity and innovativeness.
6. Expertise in computing helps you even if your primary career choice is something else.
5. Computing jobs are here to stay, regardless of where you are located.
3. Computing enables you to make a positive difference in the world.
2. Expertise in computing enables you to solve complex, challenging problems.
1. Computing is part of everything we do!

http://computingcareers.acm.org/?page_id=4
Eclipse IDE (Integrated Development Environment)

- **Editor**
- **Compiler**
- **Translator**
- **Interpreter (Virtual Machine)**

Files:
- `Hello.java`
- `Hello.class`

Computer

Users

Programmer
Eclipse IDE

```java
public class HelloWorld {
    public static void main(String[] args) {
        double degreesFahrenheit = 212.0;
        double degreesCelsius = (degreesFahrenheit - 32) * 5 / 9;
        System.out.println(degreesCelsius);
    }
}
```
Course Website

http://pages.cs.wisc.edu/~cs302/
Retrieval Practice

2 minutes - recall and write down key points from last lecture
Survey
Number Systems

Decimal
   0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Binary
   0, 1
Decimal

\[ 100 \quad 10 \quad 1 \]

\[ 3 \quad 0 \quad 2 \quad = \quad 302 \]
Binary

1 1 1 1 0 = 14
Binary

0 1 1 0 = 6

8 4 2 1
Names for Numbers of Bits

0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1

- bit
- nibble (4 bits)
- byte (8 bits)
- 2 bytes
Hexadecimal (group bits by nibble)

<table>
<thead>
<tr>
<th>Binary</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0</td>
</tr>
<tr>
<td>0001</td>
<td>1</td>
</tr>
<tr>
<td>0010</td>
<td>2</td>
</tr>
<tr>
<td>0011</td>
<td>3</td>
</tr>
<tr>
<td>0100</td>
<td>4</td>
</tr>
<tr>
<td>0101</td>
<td>5</td>
</tr>
<tr>
<td>0110</td>
<td>6</td>
</tr>
<tr>
<td>0111</td>
<td>7</td>
</tr>
<tr>
<td>1000</td>
<td>8</td>
</tr>
<tr>
<td>1001</td>
<td>9</td>
</tr>
<tr>
<td>1010</td>
<td>A</td>
</tr>
<tr>
<td>1011</td>
<td>B</td>
</tr>
<tr>
<td>1100</td>
<td>C</td>
</tr>
<tr>
<td>1101</td>
<td>D</td>
</tr>
<tr>
<td>1110</td>
<td>E</td>
</tr>
<tr>
<td>1111</td>
<td>F</td>
</tr>
</tbody>
</table>
Characters

0000 0000 0100 0001 = 0x0041 = 65

Unicode

<table>
<thead>
<tr>
<th>Unicode</th>
<th>ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>space</td>
</tr>
<tr>
<td>33 !</td>
<td>!</td>
</tr>
<tr>
<td>34 “</td>
<td>&quot;</td>
</tr>
<tr>
<td>35 #</td>
<td>#</td>
</tr>
<tr>
<td>36 $</td>
<td>$</td>
</tr>
<tr>
<td>37 %</td>
<td>%</td>
</tr>
</tbody>
</table>

http://www.ssec.wisc.edu/~tomw/java/unicode.html
Color

Red, Green, Blue (RGB)

181, 1, 1
B70101

https://umark.wisc.edu/brand/web/colors.php
Data Types

Primitive (the most common)

- **int**: whole numbers (4 bytes)
- **double**: floating point numbers (8 bytes)
- **char**: single character (2 bytes)
- **boolean**: true or false value (?)
Data Types

Reference

Where to find the data in memory
4 or 8 bytes depending on VM
What does this mean?

int age;
What does this mean?

```java
int age;
age = 25;
```
What does this mean?

```java
int age;
age = 25;
age = age + 1;
```
What does this mean?

```c
int i;
int j;
i = 1;
j = 2;
i = j;
```
Does this mean the same thing?

```c
int i;
int j;
i = 1;
i = j;
i = j;
j = 2;
```
Demonstration

Java Visualizer