

# **CS 161**

## **Intro to CS I**

What is CS all about?

# Odds and Ends



- Assignment 1 Due Sunday, 11:59pm
- Questions?



# Reflections

- What do the following Linux commands do?

– ls

– list contents of directory/folder

– mkdir

– create directory

– cd

– change directory

cd ~

- What is vi/vim?

text editor

- How do get into the insert mode? Command mode?

# 001 Code/002 will finish Friday



Oregon State University  
College of Engineering

```
2. ENGR
Re-attach Fullscreen Stay on top Duplicate
1 #include <iostream> //library
2
3 int main() {
4     std::cout << "hello everyone" << std::endl;
5
6     return 0;
7 }
-- INSERT -- 4,4 All
```

*Preprocessor  
directive*

*type  
we return*

*no error* *literal*

# 001 Lecture Demo...



Oregon State University  
College of Engineering

```
3. ENGR
Re-attach Fullscreen Stay on top Duplicate
1 #include <iostream> //library
2 #include <climits>
3 #include <cmath> //access the pow() function
4
5 using namespace std;
6
7 int main() {
8     cout << "hello everyone" << endl;
9
10    //signed mins are off by one from cplusplus.com chart!
11    cout << "climits stuff" << endl;
12    cout << "int max: " << INT_MAX << endl;
13    cout << "int min: " << INT_MIN << endl;
14    cout << "unsigned int max: " << UINT_MAX << endl;
15
16    cout << "calculated things" << endl; //calucate for int, short, long
17    //typecast real number returned from pow to type of whole number you want
18    cout << (unsigned int) pow(2,(sizeof(int)*8))-1 << endl;
19
20
21    return 0;
22 }
```

# 002 Lecture Demo...



Oregon State University  
College of Engineering

```
3. ENGR
Re-attach Fullscreen Stay on top Duplicate
1 #include <iostream> //brings in library, by preprocessor directive
2 #include <climits>
3 #include <cmath>
4
5 using namespace std;
6
7 int main() {
8     cout << "hello" << endl;
9     cout << "climits" << endl;
10    cout << "LONG_MAX: " << LONG_MAX << endl;
11    cout << "SHRT_MAX: " << SHRT_MAX << endl;
12    cout << "SHRT_MIN: " << SHRT_MIN << endl;
13    cout << "USHRT_MAX: " << USHRT_MAX << endl;
14
15    cout << "sizeof" << endl; //calucate for int, short, long
16    //typecast real number returned from pow to type of whole number you want
17    cout << (unsigned short) pow(2,sizeof(short)*8)-1 << endl;
18    cout << pow(2,sizeof(long)*8)-1 << endl; //without typecast, you get real #
19    cout << (unsigned long) pow(2,sizeof(long)*8)-1 << endl;
20
21    return 0; //there are no problems
22 }
```

# More C++



- Programming Style: please read your class style guide
  - Program Header/Description
  - Placement of { }
  - Indentation: spaces vs. tabs
- String Literal in quotations, ""
  - Not single quotes!
    - INCORRECT: `std::cout << 'Hello World';`
  - Do not span more than one line!
    - INCORRECT: `std::cout << "Hello  
World";`

## More C++

- Escape Sequences
  - Display special characters
  - Use backslash, \, before special character to print
- Examples:  

```
std::cout << "\\\"Hello World\\\"\\n\";
```
- Refer online for common escape sequences:  
<http://en.cppreference.com/w/cpp/language/escape>





# Data Type

What are you sending the function?

- What is data?
  - Information
  - Ex: `std::cout << "Hello World!" << std::endl;`
  - Literals
    - 23, 79.5, "Hello", etc.
- What is a data type?
  - Description of the kind of information
    - Primitive Data
    - User Created – (we will cover later)

# C++ Primitive Types

- char, double, float, int, long, short, bool
- Fundamental
  - **short/int/long**: whole numbers, e.g. 45, -89, 0
  - **float/double**: real numbers, e.g. 2.612, -30.5, 2.3e5
  - **char**: characters, e.g. 'A', '&', 'x', '\\'
- Signed by default, need to preface with unsigned keyword
  - **unsigned int**
  - **unsigned float**
  - **unsigned char**



# Assignment #1 Macros

- C++: <climits>
- Use MIN and MAX macros from library  
<http://www.cplusplus.com/reference/clibrary/climits/>  
(Note that the values listed are not the values on our system!!!)
  - INT\_MAX
  - INT\_MIN
  - LONG\_MAX
  - LONG\_MIN
  - SHRT\_MAX
  - SHRT\_MIN
- Remember unsigned too...

# What is an expression?

- Set of operations producing a value

$$12 * 4 + 6 * 10$$

$$((12 * 4) + 6) * 10$$



# Pieces of an Expression

- **Operators:** indicate operation
  - Add +
  - Subtract -
  - Multiply \*
  - Divide /
  - Remainder %
- **Operands:** values in the expression
- **Evaluation:** process of obtaining results from operations on operands

# Arithmetic

- **Integer Arithmetic**

```
std::cout << 3/8; /*prints 0*/  
std::cout << 34/5; /*prints 6*/
```

- **Floating Point Arithmetic**

```
std::cout << 34.0/5.0; /*prints 6.8*/  
std::cout << 3.0/8; /*prints .375*/  
std::cout << 3/8.0; /*prints .375*/
```



# Type Casting

- **Casting**

```
std::cout << 34 / (int) 5.0; /*prints 6*/
```

```
std::cout << (int) (34 / 5.0); /*prints 6*/
```

```
std::cout << (float) 34 / 5; /*prints 6.8*/
```

- **What is wrong with these?**

```
std::cout << (int) 34 / 5.0; /*prints 6.8*/
```

```
std::cout << (float) (34/5); /*prints 6.0*/
```

# Precedence

- What is precedence?
  - Binding power of operator
  - ( $*$ ,  $/$ ,  $\%$ ) vs. ( $+$ ,  $-$ )
- How do we override precedence?
  - Parenthesis!
- Examples:
  - $12 * 4 + 6 * 10$  vs.  $((12 * 4) + 6) * 10$



# Size of Things Demo...



Oregon State University  
College of Engineering

