CS 161
Intro to CS I

C++ Strings, Errors, Debugging, and Begin Functions
Odds and Ends...

• Last week to demo Assignment 1!
• Sign up for demoing Assignment 2 early!!
• Work on Assignment #3 Design due Sunday!!!
```cpp
#include <iostream>
#include <string>  // location of C++ string object
#include <cstdlib> // location of atoi()/ASCII to integer

using namespace std;

int main() {
    string mssg="hello";  // can assign a string literal to string object
    cout << "Enter a number: ";
    cin >> mssg;
    int i; // declare it outside the for because we use if after the for
    for(i=0; i<mssg.length(); i++) {
        // if a character is not within the range of '0' - '9' on ASCII chart,
        // then it isn't a positive integer
        if(!(mssg.at(i) >= '0' && mssg.at(i) <= '9')) {
            cout << "bad input" << endl;
            break;
        }
    }
    if(i==mssg.length()) // if all the characters are digits in string, then...
        cout << atoi(mssg.c_str()) << endl; // change it to an integer
    return 0;
}
```
Documentation

• http://cplusplus.com/
Programming Errors

• Syntax errors
  – Misuse of C++ language
  – How are they caught?

• Logic errors
  – Doesn’t perform task correctly (aka. bugs)
  – How are they caught?

• Runtime errors
  – Stops your program from running
  – How are they caught?
Syntax Error Examples

• Missing main function
• Use of identifier not declared
• Misspelled Words
• Forget a Semicolon
• Forget Required Keyword
• Missing quote, curly brace, and parenthesis
• Use of single quotes instead of double
Logic Error Examples

- Poorly written programs
  - Add instead of subtract (incorrect operation)
  - Using last two digits for date
  - Same error message for different errors
  - Program that never ends
  - Add one to the largest integer

OSU Oregon State University
Runtime Error Examples

• Open a file that doesn’t exist
• Segmentation fault/Core dump
  – Infinite loop that eats memory
  – Divide by variable that is zero
  – Access memory bad memory
Debugging Errors

• Syntax:
  – **READ compiler errors** (pay attention to line #)
  – Use **google** to search for error

• Logic/Runtime
  – Use **std::cout** to find where the code is breaking
    • Print variable values
    • Print indicator messages
  – **Trace** through the code
  – **Comment** out code
Buggy Code Demo...

```cpp
#include <iostream>
#include <string>
using namespace std;

int main() {
    int num, digit, sum=0;
    string cont="y";

    cout << "Enter a 0-12 digit number: ";
    cin >> num;

    while(cont=="y") {
        //sum the digits
        for(int i=0; i<12; i++) {
            digit=num%10;
            num/=10;
            sum+=digit;
        }
        cout << "sum is: " << sum << endl;

        cout << "do this again (y-yes, n-no): ";
        cin >> cont;
    }

    return 0;
}
```
```cpp
#include <iostream>
#include <string>
using namespace std;

int main() {
    long num, digit, sum=0;  //make sure you have enough bits for number
    string cont="y";

    while(cont=="y") {
        cout << "Enter a 0-12 digit number: ";
        cin >> num;  //make sure you get the number each iteration
        cout << "num is: " << num << endl;  //print number for debugging
        //sum the digits
        for(int i=0; i<12; i++) {
            digit=num%10;
            num/=10;
            sum+=digit;
        }
        cout << "sum is: " << sum << endl;

        cout << "do this again (y=yes, n=no): ";
        cin >> cont;
        sum=0;  //make sure sum is set back to zero
    }

    return 0;
}
```