CS 161
Intro to CS I
Continue Functions
Odds and Ends...

• Design Critique due!!!
• Exam I – Friday, 10/21
• Demo Assignment 2 this week!!!
void Functions

• Doesn’t return a value
• Still has arguments/parameters

Get into groups of 4-5
• Write a void check_positive_int ()
  – What does the parameter look like?
  – What does the function call look like?
  – Is it more useful to return a value?
```cpp
#include <iostream>
#include <string>  // location of C++ string object
#include <cstdlib>  // location of atoi()/ASCII to integer
using namespace std;
// check to make sure you have a good positive int
void check_pos_int(string);

int main() {
    string mssg="hello";  // can assign a string literal to string object
    cout << "Enter a number: ";
    cin >> mssg;
    check_pos_int(mssg);

    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;
}

void check_pos_int(string 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;
            exit(0);  // get out of the whole program
        }
    }
    return 0;
}
```
Scope (Visibility)

- Part of program in which a declaration is valid
- Local variable
  - Declared inside a function only accessible inside function
- Localizing variables
  - Declaring variable in innermost scope
C++ Pass by Value

void swap(int, int);

int main() {
    int a=5, b=10;
    swap(a, b);
    cout << "a: " << a << " b: " << b;
}

void swap(int x, int y) {
    int temp = x;
    x = y;
    y = temp;
}

• What if we didn’t have temp?
More about break, exit, and return

- **break** – used with switch and loops, breaking out of the closest associated case or loop (for, while, or do while). **This statement can only occur in a loop or case**, otherwise the compiler yells!

- **return** – leave the current function, which exits the program when in the main() function. **You can put this anywhere inside any function**, otherwise the compiler yells!

- **exit()** – exit the entire program, no matter where this is encountered. **You can put this anywhere inside any function**, **as long as you include <cstdlib>**, otherwise the compiler yells!
Programming Demo

• Would it be more helpful to return a value from check_positive_int()?
```cpp
#include <iostream>
#include <string>   // location of C++ string object
#include <stdlib>   // location of atoi() ASCII to integer
using namespace std;

// check to make sure you have a good positive int
int check_pos_int(string);

int main() {
    string mssg="hello";  // can assign a string literal to string object
    cout << "Enter a number: ";
    cin >> mssg;
    cout << check_pos_int(mssg) << endl;

    return 0;
}

// return a good int when we know we have one
int check_pos_int(string mssg){
    int i;  // declare it outside the for because we use if after the for
    do {
        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()) {
            cout << "Enter a number: ";
            cin >> mssg;
        } else
            return atoi(mssg.c_str());
    } while(1);
}  ```
More About Functions

• Do not use global variables!

• Function Headers
  – Description, Parameters, and Return Value
  – Postconditions
    • What is this?
  – Preconditions
    • What is this?
Even though it works,
DO NOT USE GLOBAL VARIABLES!!!