CS 161
Intro to CS I

More About Functions:
Scope, Return Values, Default Values, and Overloading
void Functions

• Doesn’t return a value
• Still has arguments/parameters
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
Illegal access outside loops

```c++
for(x = 0; x < 10; x++) {
    int y = 10;
    cout << "The value of x * y is: " << x*y << endl;
}
cout << "The value of y is: " << y << endl; /*y outside scope*/
```

- How do we fix this?
- What about if/else blocks?
Illegal access in functions

```c
int main () {
    int x=2, y=3;
    compute_sum();
    sum = x+y;  //error: sum hasn’t been declared
    return 0;
}

void compute_sum() {
    int sum = x+y;  //error: x and y outside scope
}
```
Arguments/Parameters Demo
Global Variables
Do NOT use them!!!
Returning Values Demo...
Back to **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!
More About Functions

• Do not use global variables!

• Function Headers
  – Description, Parameters, and Return Value
  – Preconditions
    • What is this?
  – Postconditions
    • What is this?
Default Args

#include <iostream>

using std::cout;
using std::endl;

int pwr(int, int n=1); // Example of default args

int main() {
    int base=2, expn=8;

    cout << "The power function: " << pwr(base, expn) << endl;
    cout << "The power function: " << pwr(base) << endl;

    return 0;
}

int pwr(int x, int n) {
    int num=1;

    for(int i=0; i < n; i++) {
        num*=x;
    }

    return num;
}
C++ Function Overloading

• Multiple functions w/ same name
• Arguments determine function
• Default Args can be done w/ overloading
• Example: pow()