2 Predefined Functions

- C++ comes with libraries of predefined functions
- Example: **sqrt** function
  - the_root = sqrt(9.0);
  - returns, or computes, the square root of a number
  - The number, 9, is called the argument
  - the_root will contain 3.0

2.1 Function Calls

- sqrt(9.0) is a function call
  - It invokes, or sets in action, the **sqrt** function
  - The argument (9), can also be a variable or an expression
- A function call can be used like any expression
  - ```
    bonus = sqrt(sales) / 10;
    ```
  - ```
    cout << "The side of a square with area " << area << " is " << sqrt(area);
    ```

2.2 Function Call Syntax

- Function_name (Argument_List)
  - Argument_List is a comma separated list:(Argument_1, Argument_2, . . . , Argument_Last)
- Example:
  - ```
    side = sqrt(area);
    cout << "2.5 to the power 3.0 is " << pow(2.5, 3.0);
    ```

2.3 Function Libraries

- Predefined functions are found in libraries.
- The library must be "included" in a program to make the functions available.
- An include directive tells the compiler which library header file to include.
- To include the math library containing **sqrt**():
  ```
  #include <cmath>
  ```
- Newer standard libraries, such as **cmath**, also require the directive
  ```
  using namespace std;
  ```

2.4 Other Predefined Functions

- Math Functions `http://www.cplusplus.com/reference/cmath/`
- C Standard General Functions `http://www.cplusplus.com/reference/cstdlib/`
- Many others in C++ standard
- Many third party provides libraries, ex: `http://www.boost.org`, `http://eigen.tuxfamily.org/index.php?title=Main_Page`
  ```

2.5 Type Casting

- Recall the problem with integer division:

```
int total_candy = 9, number_of_people = 4;
double candy_per_person;
candy_per_person = total_candy / number_of_people;
```

  - candy_per_person = 2, not 2.25!
  - Integer division occurs before type conversion.

- A Type Cast produces a value of one type from another type
  - **static_cast<double>(total_candy)** produces a double representing the integer value of total_candy

2.5.1 Type Cast Example

-
```
int total_candy = 9, number_of_people = 4;
double candy_per_person;
candy_per_person = static_cast<double>(total_candy) / number_of_people;
```

  - candy_per_person now is 2.25!

- This would also work:

```
candy_per_person = total_candy / static_cast<double>(number_of_people);
```

2.5.2 Old Style Type Cast

- C++ is an evolving language
- This older method of type casting may be discontinued in future versions of C++

```
candy_per_person = (double)total_candy / number_of_people;
```