Lecture 6

Chapter 3.1
Topics

- Argument
- Value return
- Function call
- `#include` directive
- Some predefined functions
- Predefined `void` functions
- Exit function
- `rand`
- `Seed`
- Floating point random numbers
Argument

• Argument
  – Is the value the function starts out with.
  – An example is a function called squareVolume would take as input the length of a side.
  – squareVolume(3);
    • Where 3 is an integer value of the length of a side.
    • 3 is the argument.
    • It could also be a variable.
Value return

• This is the value a function returns.
  – This is defined in the function definition.
  – Example is main
    • int main
      – returns an int.
  – Each function can only return one value.
    – int squareVolume(side) would return the value of (side * side)
Function call

• This is how a function is called
  – theRoot = sqrt(9.0);
    • This calls the function sqrt with an argument of 9.0 and assigns the value it returns to the variable theRoot.
#include directive

• Include directive is how premade functions in the C++ library are accessed for use in your program.
  – Example
    • #include <cmath>
Some predefined functions

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Arg Type</th>
<th>Return Type</th>
<th>Example</th>
<th>Value</th>
<th>Library Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>sqrt</td>
<td>Square root</td>
<td>double</td>
<td>double</td>
<td>sqrt(4.0)</td>
<td>2.0</td>
<td>cmath</td>
</tr>
<tr>
<td>pow</td>
<td>Power</td>
<td>double</td>
<td>double</td>
<td>pow(2.0,3.0)</td>
<td>8.0</td>
<td>cmath</td>
</tr>
<tr>
<td>abs</td>
<td>Absolute value for int</td>
<td>int</td>
<td>int</td>
<td>abs(-7)</td>
<td>7</td>
<td>cstdlib</td>
</tr>
<tr>
<td>fabs</td>
<td>Absolute value for float</td>
<td>double</td>
<td>double</td>
<td>fabs(-7.5)</td>
<td>7.5</td>
<td>cmath</td>
</tr>
<tr>
<td>floor</td>
<td>Round down</td>
<td>double</td>
<td>double</td>
<td>floor(3.2)</td>
<td>3.0</td>
<td>cmath</td>
</tr>
<tr>
<td>exit</td>
<td>End program</td>
<td>int</td>
<td>void</td>
<td>exit(1);</td>
<td>None</td>
<td>cstdlib</td>
</tr>
<tr>
<td>ceil</td>
<td>Round up</td>
<td>double</td>
<td>double</td>
<td>ceil(3.01)</td>
<td>4.0</td>
<td>cmath</td>
</tr>
</tbody>
</table>
Predefined void functions

• These functions do no return anything.
• They usually perform an action.
  – Examples
    • exit(1);
Exit function

• This program ends the function immediately.
• It returns void to the program.
• It returns the argument to the operating system.
• exit(0) by default and exit(1) for an error.
rand

• Is used as a “random” number generator that generates number from zero to RAND_MAX.
• RAND_MAX sets the upper bound of the numbers generated.
rand scaling

• Scaling can be done like this
• for(int i = 0; i<10; i++)
    
    cout << (rand() % 11) << endl;
– The symbol % is used to get the remainder of rand() / 11 therefor the values will always be from 0 to 10.
Pseudorandom numbers

• rand doesn’t actually generate completely random numbers, they only appear to be random.

• The sequence of pseudorandom numbers is usually determined by one number, the seed.

• srand sets the value of the seed

srand(10);

for(int i = 0; i<10; i++)
    cout << (rand() % 11) << endl;
Floating point random numbers

• The following is how to make a floating point pseudorandom generator.

\[(\text{RAND\_MAX} - \text{rand}()) / \text{static\_cast<double>}(\text{RAND\_MAX})\]