CS 111
- Kangmei Yang
Functions:

Why do we need functions?
- A task might be performed repeatedly throughout different parts of the program
- Instead copy and paste the same code, we create this part as a function, whenever this code is needed in the program, we make a function call.
- It’s easier to make modification in one place than multiple places.
- Keep main function simple and easier to manage while focusing on the task.
Functions:

```c
#include <cmath>
sqrt(5.5);
- takes a double, return a double, the square root of the given double

#include <cstdlib>
rnd();
- doesn’t take any arguments, gives an random calculated integer.
```
Declare and Define Function:

- Prototype  This gives the compiler a preview of what your function would look like
  • After “using namespace std;”, before “int main()”

- Definition  Define the actions the function should take
  • Usually after “int main{ ... return 0;}”
Prototype Model:

return_type function_name( parameter_list);
- return_type the data type of which the function returns
- function_name name of the function
- parameter_list List of data type, along with name of parameter(s)
return_type:

The function could return 1 or none object back to the caller.

return_type specify the data type of this object

Ex:
sqrt(); // returns double
rand(); // returns int

If the function doesn’t return anything, the return type would be “void”.
Parameter list:

The function might need some information/data in order to finish the task.

Pass all the data as it’s parameter
Ex: sqrt(100.1); // take a double value to compute it’s square root.

If the no information is needed, the parameter list is empty.

If more than one information is needed, each separated by a comma “,,”

Each parameter has it’s own data type and name!