Chapter 6
1. Where do you define parameter variables?
2. When a function accepts multiple arguments, does it matter in what order the arguments are passed?
3. How do you return a value from a function?
4. What is the advantage of breaking your application’s code into several small procedures?
5. A program contains the following function:

```c++
void display(int arg1, double arg2, char arg3){
    cout << “Here are the values: “
        << arg1 << “ “ << arg2 << “ “
        << arg3 << endl;
}
```

Write a statement that calls this function and passes the following variables to it:
```c++
int age;
double income;
char initial;
```
6. The ____________ is the part of a function definition that shows the function name, return type, and parameter list.
7. If a function doesn’t return a value, the word ____________ will appear as its return type.
8. Either a function’s ____________ or its ____________ must precede all calls to the function.
9. Values that are sent into a function are called ____________.
10. When only a copy of an argument is passed to a function, it is said to be passed by ____________.
11. A(n) ____________ eliminates the need to place a function definition before all calls to the function.
12. The ____________ statement causes a function to end immediately.
13. Reference variables are defined like regular variables, except there is a(n) ____________ in front of the name.
14. Reference variables allow arguments to be passed by ____________.
15. Two or more functions may have the same name, as long as their ____________ are different.