Answers to Self-Check Exercises

1. answer: int Scale (int name1, double name2);

2. answer: double Blend();

3. answer: void Dither (double name1, double name2, float name3);

4. answer: #include “stats.h”

5. answer: #include <iostream>

6. answer:
   #ifndef NO_MORE_Count_AND_Average
   int Count = 0;
   float Average = 1.0;
   #define NO_MORE_Count_AND_Average
   #endif

7. answer: The statement makes all declarations in the std namespace accessible without having to use the namespace name and the scope resolution operator ::.

8. answer: std::cout << “Fatal Error”;

9. answer: cerr, clog

10. answer: cout << oct << HeadCount << endl;

11. answer: This means that all numbers are expressed in the requested base for the affected stream until another base change is requested.

12. answer:
    10
    3156

13. answer:
    123456789
    Hello
    Good bye

14. answer: $$$$3
15. answer: 123456789 $\$z \$ $z \$

16. answer: 123456789 $\$ z\$ $ z\$

17. answer: 1

18. answer: 10
31 56

19. answer: +16

20. answer: 020

21. answer: rand()

22. answer: srand()

23. answer:
```cpp
#include <iostream>
#include <string>
#include <stdlib.h>
#include <time.h>
using namespace std;

int main() {
    srand((unsigned int) time(0));
    int heads = 0;
    for (int i=1; i<= 1000; ++i) {
        if (rand() % 2 == 1) {
            ++heads;
        }
    }
    cout << "Number of heads = " << heads << endl;
    return 0;
}
```
24. answer:
#include <iostream>
#include <fstream>
#include <string>
using namespace std;

int main() {
    ifstream fin("data.txt");
    int value;
    int NbrOfOddIntegers = 0;
    while (fin >> value) {
        if (value % 2 == 1)
            ++NbrOfOddIntegers;
    }
    cout << "There are " << NbrOfOddIntegers
         << " odd number(s) in the file data.txt." << endl;
    return 0;
}

25. answer:  assert(Count);

26. answer:  assert(Index(n) < 0);