

1. What is the size of file1, in bytes, after the code below is run in Windows? **ANSWER: 7**
 What is the size of file2? **ANSWER: 6** (5 pts each)

```
ofstream out1("file1");
ofstream out2("file2",
    ios::out | ios::binary);
out1 << "hello\n";
out2 << "hello\n";
```

2. Write a program `countletter.cpp` that takes a filename and a single character on the command line and prints the number of occurrences of that character in the file (case sensitive). The program should print an error message if there are not exactly two command line arguments or if the file is not found. For example, if `foo.txt` contains "this is a test", then (30 pts):

```
countletter foo.txt t
foo.txt contains 3 of t
countletter bar.txt x
bar.txt not found
countletter
2 arguments expected
```

```
// ANSWER
#include <iostream>
#include <fstream>
using namespace std;
int main(int argc, char **argv)
{
    if (argc != 3)
        cout << "2 arguments expected\n";
    else
    {
        ifstream in(argv[1]);
        if (!in)
            cout << argv[1] << " not found\n";
        else
        {
            int count = 0;
            char c;
            while (in.get(c))
                if (c == argv[2][0])
                    ++count;
            cout << argv[1] << " contains "
                << count << " of "
                << argv[2][0] << "\n";
        }
    }
    return 0;
}
```

3. Write a function `avg()` that takes a `map<string, double>` using pass by reference, and returns the average of the values. If the map is empty, return 0. For example (30 pts):

```
map<string, double> m;
cout << avg(m); // 0
m["cat"] = 3.0;
m["dog"] = 4.2;
cout << avg(m); // 3.6
```

```
// ANSWER
double avg(map<string, double>& m)
{
    double sum = 0;
    map<string, double>::iterator p;
    for (p = m.begin(); p != m.end(); ++p)
        sum += p->second;
    if (int(m.size()) == 0)
        return 0;
    else
        return sum / int(m.size());
}
```

4. Write a function `sum()` that takes an array of `int` and its size and returns the sum of the elements, for example (30 pts):

```
int a[5] = {10, 20, 30, 40, 50};
cout << sum(a, 5); // 150
cout << sum(a+1, 3); // 90
```

```
// ANSWER
int sum(int *a, int n)
{
    int result = 0;
    for (int i = 0; i < n; ++i)
        result += a[i];
    return result;
}
```