I/O in C++

Reading:

- Appropriate sections of your C++ book.
  E.g.,
  Lippman, chapter 20, or
  Stroustrop chapter 21

#include <fstream>
#include <string>

int main(void)
{
    // Output to "the standard output".
    cout << "Hello.\nHow old are you? ";

    // Input from "the standard input".
    int age;
    cin >> age;
    cout << "Wow, " << age << ". You're that old?\n";

    // Output to a file (as text).
    fstream fs;
    fs.open ("outfile", ios::out);
    fs << age;
    fs.close();

    // Input from a file (as text).
    int num;
    fs.open ("infile", ios::in);
    fs >> num;
    cout << "I just read from a file: " << num << ".\n";
    fs.close();
/ Output to a file (binary).
fs.open("binfile", ios::out | ios::bin);
  char c = 'A';
  fs.write(&c, sizeof(char));

  for (int i=0; i < 10; i++)
    fs.write(&i, sizeof(int));
fs.close();

// Input from a file (binary).
fs.open("binfile", ios::in | ios::bin);
fs.read(&c, sizeof(char));
  cout << "I read the character: " << c << "." << endl;
fs.read(&num, sizeof(int));
  cout << "I read the integer: "
    << num << ".\n";
fs.read(&num, sizeof(int));
  cout << "The rest of the binary file contains:\n";

// Input until end of file.
while (!fs.fail()) {
  cout << "  " << num << endl;
    fs.read(&num, sizeof(int));
}

  cout << "For the record, sizeof(int) is "
    << sizeof(int) << "," << endl;
  cout << "For the record, sizeof(char) is "
    << sizeof(char) << "," << endl;
return 0;

qew.cs 1% cat infile
12345
Hello, here's a line
and here is another line.
Let's put some more
stuff in here so that
we don't hit end-of-file before we're
doing trying to read input.
qew.cs 2% a.out
Hello.
How old are you? 3
Wow, 3. You're that old?
I just read from a file: 12345.
I read the character: A.
I read the integer: 0.
The rest of the binary file contains these integers:
1
2
3
4
5
6
7
8
9
For the record, sizeof(int) is 4.
For the record, sizeof(char) is 1.
dvp.cs 96% cat outfile
3
```cpp
#include <fstream>
#include <string>

int main(void)
{
    // Input from a file (text), grabbing a line at a time.
    ifstream in;
in.open("infil", ios::in);

    for (int i = 1; i<=5; i++) {
        char a[100];      // Yuk: magic numbers.
in.getline(a, 100, '\n');
cout << "As a C-style string: " << a << "\n";

        // Making a C++ string out of this C-style string a.
        string s(a, 0, 99);
cout << "As a C++ string : " << s << "\n";
    }
    return 0;
}
```
qew.cs 849% cat infile
12345
Hello, here's a line
and here is another line.
Let's put some more
stuff in here so that
we don't hit end-of-file before we're
doing trying to read input.

qew.cs 850% a.out
As a C-style string: 12345
As a C++ string : 12345
As a C-style string: Hello, here's a line
As a C++ string : Hello, here's a line
As a C-style string: and here is another line.
As a C++ string : and here is another line.
As a C-style string: Let's put some more
As a C++ string : Let's put some more
As a C-style string: stuff in here so that
As a C++ string : stuff in here so that