

Sending Output to a File

Sending output to a file requires some extra steps in your program. The program shown below uses both file I/O and stream I/O.

Example 1:

```
#include <iostream>
#include <fstream> // needed to use a file
using namespace std;
int main()
{
    int x,y;
    ofstream outfile("p2output.txt"); // file declaration declares outfile
                                     // as internal name. Creates, names,
                                     // & opens disk file p2output.txt
                                     // and associates it with outfile

    cout << "This line goes to the screen" << endl;
    outfile << "This line goes to the output file" << endl;

    x = 10;
    cin >> y;

    cout << x << " " << y << endl; // writes ONLY to the screen
    outfile << x << " " << y << endl; // writes ONLY to the file

    outfile.close(); // closes file
    return 0;
}
```

Location of Declaration

Declaring the file may be done inside the main program as shown in these examples, or it may be done above main. **Declaring the file above main makes the file available for use in the main program as well as in other functions the program uses (see Chapter 5).*** The file must be opened before the first use of the file. Do not open a file more than once (unless you close it in between), because each call to open() starts reading or writing over again at the beginning of the file. *The part in bold is critical for Assn 3, which will be based on Chapters 4 and 5.

Example 2:

Here are the steps needed to send output to file "p3.out"; the file declaration is above main:

```
#include <iostream>
#include <fstream>
using namespace std;
ofstream outfile("p3.txt"); // open file "p3.txt" and call it outfile
int main()
{
    int num = 15;

    outfile << num; // write the value of num to the file
    outfile.close(); // close the file
    return 0;
}
```