

Feb 06, 09 15:53

filename\_test.cpp

Page 1/2

```
// file: filename_test.cpp
//
// Test program to use C++ strings to make filenames in various ways
//
// Programmer: Dick Furnstahl  furnstahl.1@osu.edu
//
// Revision history:
// 20-Feb-2004  original version, for 780.20 Computational Physics
// 06-Feb-2009  minor upgrades
//
// Notes:
// * compile with: "g++ -Wall -o filename_test filename_test.cpp"
// * note that we include <string> for strings and <sstream> for
//   building string streams
// * building string stream objects is just like doing output using
//   iostream and iomanip, with <<'s and manipulators
//
//*****
// include files
#include <iostream>    // cout and cin
#include <iomanip>     // manipulators like setprecision
#include <fstream>    // file input and output
#include <string>     // C++ strings
#include <sstream>    // C++ stringstream class (could omit iostream)

using namespace std;

//*****
int
main (void)
{
    // first a little test of string input from the console
    // notice that we don't specify the length of the string
    string test_string = ""; // initialize teststring to null

    while (true)        // do forever (until we hit a "break" statement)
    {
        cout << "Enter a single word (\\"quit\\" to quit): "; // note the \
        // you can input strings using cin just as with any other data type
        cin >> test_string;

        if (test_string == "quit") // comparing strings with ==
        {
            cout << "Ok, we'll move on ..." << endl << endl;
            break; // jump out of the while loop
        }

        // strings are output just like other data types
        cout << "You entered: " << test_string << ". didn't you?" << endl;
    }

    //*****
    // second, we'll concatenate two strings using "+"
    string first_string = "", second_string = "";

    cout << "Enter two words: ";
    cin >> first_string >> second_string;

    cout << "Combined we get " << first_string + second_string << endl << endl;

    cout << "Type any word to continue: ";
    cin >> first_string;
    cout << endl << "*****" << endl;

    //*****
    // next, open a file with a fixed string
```

Feb 06, 09 15:53

filename\_test.cpp

Page 2/2

```
string my_filename1; // declare the string
my_filename1 = "test.out"; // test name

cout << "Try opening the file " << my_filename1 << " for output." << endl;

// open an output file stream
// [you could combine these as: ofstream file1 (my_filename1.c_str());]
ofstream my_file1;
// use .c_str() to convert the string filename1 to char * type
my_file1.open (my_filename1.c_str());

my_file1 << "This is a test of " << my_filename1 << endl;
my_file1 << "The length of the filename is " << my_filename1.length();

my_file1.close(); // close the output file

cout << "Look at " << my_filename1 << " to see if this worked."
    << endl << endl;

//*****
// next, create a string with the stringstream class
ostringstream my_filename_stream; // declare a stringstream object

// first, put an integer in the filename
int i = 3;
// you can load the string from a stream just like output streams
my_filename_stream << "test_stream" << i << ".out";
// use .str() to convert to a string
string my_filename2 = my_filename_stream.str();

cout << "The output filename is " << my_filename2 << endl;

ofstream my_file2; // now for a filename
my_file2.open (my_filename2.c_str()); // use .c_str() to convert to a char*

my_file2 << "This is a test of putting the number " << i
    << " in this filename!" << endl;

my_file2.close(); // close the output file
cout << "Look at " << my_filename2 << " to see if this worked."
    << endl << endl;

// now try a float in the filename
my_filename_stream.str (""); // clear the string stream
float x = 3.14159; // more digits than we plan to use
my_filename_stream << "test_stream" << "_x" << setprecision (2) << x
    << ".out";

// don't introduce an intermediate string for the name this time
cout << "The output filename is " << my_filename_stream.str() << endl;
// use .str() to convert to a string, then .c_str() to convert to a char *
ofstream my_file3; // new filename (could have re-used my_file2)
my_file3.open (my_filename_stream.str().c_str());

my_file3 << "This is a test of putting two digits of the float " << x
    << " in this filename!" << endl;

my_file3.close(); // close the output file
cout << "Look at " << my_filename_stream.str()
    << " to see if this worked." << endl << endl;

return (0);
}
```