Reading from and Writing to Files

Section 3.12 & 13.1 & 13.5

Files (3.12)

- Data stored in variables is temporary
- We will learn how to write programs that can
  - Create files
  - Write to files
  - Read from files

Steps to Using Files

- There are five steps that must be taken in order to use files in C++
  1. Include header files
  2. Define a file stream object
     - variable to represent a file
  3. Open the file
  4. Use the file
  5. Close the file

1. Libraries

- To access files you will need to include
  - <iostream>
  - <fstream>

2. File Stream Objects (Variable)

```
ifstream inputFile;
ofstream outputFile;
fstream inAndOut;
```

- File stream objects are the ways that you refer to the files you are using
  - Can specify which input/output file to use
  - May input from more than one file
  - May output to more than one file

3. Opening Files

```
inputFile.open("filename")
```

- Same syntax for both input and output files
- Filename is a string literal
- Example:

```
ifstream inputFile;
inputFile.open("input.dat");
```

Check File Opened Correctly

- Make sure that it opened correctly

```
if(inputInfo != true)
{
    cout << "Error opening input file ";
    exit(1);
}
```


4. Using File Streams

- Use input file variable wherever you use cin
- Examples:
  - `inputFile >> num;`
- Output output file variable wherever you use cout
- Examples:
  - `outputFile << num;`

Example: Writing to a File

- The following program asks the user to input numbers and writes these numbers to a file

Example

```
#include<fstream>
#include<iostream>
using namespace std;
int main()
{
   ofstream outputFile;
   int num;
   outputInfo.open("out.dat");
   if(!outputInfo)
   {
       cout << "*** Error opening file" << endl;
        exit (1);
   }
   cout << "Enter a number (9999 to quit): ";
   cin >> num;
   while(num != 9999)
   {
        outputFile << num << " ";
        cin >> num;
   }
   outputInfo.close();
   return 0;
}
```


Reading from a File

- Write a program that will read in a sequence of numbers (double) from a file and calculate the sum. Assume that the last number is the trailer (-9999)

Reading Until the EOF (p 811)

- It is possible to read from a file until the end is reached

```
while(inputFile >> num)
{
     cout << num << " ";
     sum += num;
}
```

Reading Characters

- Write a program that reads in some text from a file and outputs that text to the screen
- The file contains:

```
Hello Everyone!
I'm a file that
contains some text.
```


Solution

```
ifstream inputFile;
char letter;

inputFile.open("in.dat");
if (!inInfo)
{
    cout << "*** Error opening file" << endl;
    exit (1);
}

while (inputFile >> letter)
{
    cout << letter;
}
cout << endl;
inputFile.close();
```

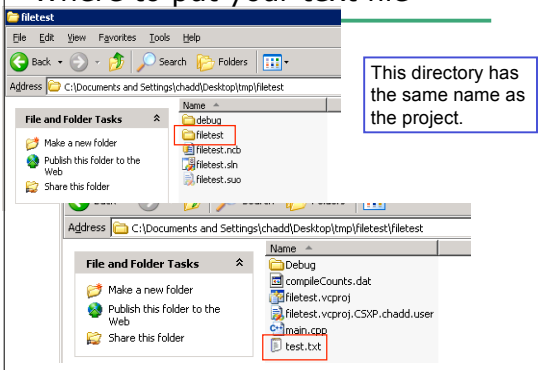

The Output (p 828)

- `HelloEveryone!I'mafilethatcontainssometext.`
- What's happened?!
- All spaces, tabs, and new lines have been ignored.
- This is because >> only reads visible characters
- How can we read all characters so that the output looks exactly like the input

Solution

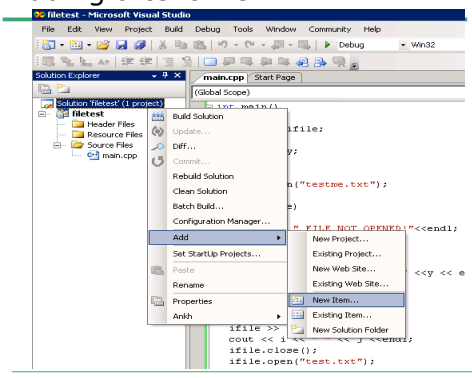
```
ifstream inputFile;
char letter;

inputFile.open("in.dat");
if(!inInfo)
{
    cout << "*** Error opening file" << endl;
    exit (1);
}

while(inputFile.get(letter))
{
    cout << letter;
}
cout << endl;
inputFile.close();
```

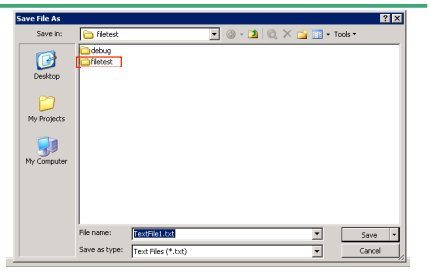

Where to put your text file



This directory has the same name as the project.

Adding a text file





- Be sure you “Save As” and save the file into the correct directory

Practice

- Write a program that will read the following file and find the largest value. The file may contain any number of values.
- Output the largest value to the screen and to the file largest.txt.
- File:

```
59
98
99
77
66
73
85
```


Practice

- Read in the following file. The rows represent students, the columns represent assignments. There are 5 assignments and an unknown number of students. Find the average grade for each student and for each assignment. Print the results to the screen and the file “results.txt”

```
99.0 98.0 89.5 77.5 66.0
73.0 85.0 77.5 89.5 89.0
62.0 84.0 83.0 77.0 88.5
```

Problem

- Consider the data file below, where - indicate spaces:

```
--12--33.4
-d--12.3
-2--5
```

- What values would be assigned to the variables for each of the statements below where `inputFile` is the file variable?

```
int i,j;
double x,y;
char ch;
```

  - `inputFile >> i >> x >> y;`
  - `inputFile >> i >> j;`
  - `inputFile >> ch >> i;`
  - `inputFile >> x >> y >> ch >> x;`