



Read/Write Files

Presented By: Josh

Files

- An alternative from using Terminal
- Read data from files and write to files
- Works similar to `cout` and `cin`

Reading Files

- `#include <fstream>`
- Must first instantiate variable with type `fstream`, e.g. :
`fstream myfile;`

Reading Files cont'd

- To open an existing file :

```
fstream myfile;
```

```
myfile.open("Filename.txt")
```

- Reading is almost identical to cin except use variable name.

```
string words;
```

```
myfile >> words; //Use this to get a word(separated by a  
recurring character, generally a space)
```

```
getline(myfile, words); //Use this to get a whole line
```

Writing to Files

- Must use : `#include <fstream>`
- Still need to instantiate a variable of type `fstream`:
`fstream otherfile;`
- Do it the same way you would with `cout`:
`otherfile << "This is text" << endl;`

Types of Writing

- Append:
 - Adds on to the end of whatever file you are writing to
 - Use `.open("Textfile.txt", ios::app | ios::in | ios::out)`
- Overwrite:
 - Default option when using `.open("Textfile.txt")`
 - As you insert into a file, it replaces what was there as you type
 - Alternatively, you can use `.open("Textfile.txt", ios::trunc | ios::in | ios::out)`
 - This, or `.clear()` will delete the contents of the file either when or after it opens

Helpful Functions

- `.eof()` : returns true if you are at the end of the file.
- `.close()` : closes the file. ALWAYS CLOSE FILES YOU HAVE OPENED!
- `.is_open()` : returns true if file in question is open.
- `.clear()` : deletes the contents of a file