C++:
Streams (specifically file I/O)

PLEASE Look at this Video

- 3 minute video on pointers and references
- https://www.youtube.com/watch?v=LwAPKuJpvmA
- (BTW I'm sorry about its weird background music)

File I/O

- #include <fstream>
- Provide char I/O to files
- Usually want to follow pattern

  open ->read or write->close

Reading file use ifstream

Writing file use ofstream

See http://www.cplusplus.com/doc/tutorial/files/

File I/O Opening a file

- .open(filename,mode)
- Filename is const char *, *a pointer*.
- Use std::string instead, easier and safer
- String.c_str() will return a const char pointer to string contents
- mode (flags)

ios::inOpen	**for input**
ios::outOpen	**for output**
ios::binaryOpen	**binary, encryption, images etc**
ios::ate	**Set initial position at the end of the file. Default is beginning**
ios::app	**append to end of the file**
ios::trunc	**if exists, open and delete contents**

File I/O reading a file

```
std::string myfile = "MyTestFile.txt";


void readFile(){
    ifstream myInputfile;
    myInputfile.open(myfile.c_str(), ios::in);
    //note the .c_str() call on MYFILE
    //read and count the data

    if (myInputfile.is_open()){
        std::string line;

        while (!myInputfile.eof()) {              //exits when reach end of file
            getline(myInputfile, line);           //gets a line up to '/n' char
            cout<<line;
        }
        myInputfile.close();
    }
}
```



See http://www.cplusplus.com/doc/tutorial/files/

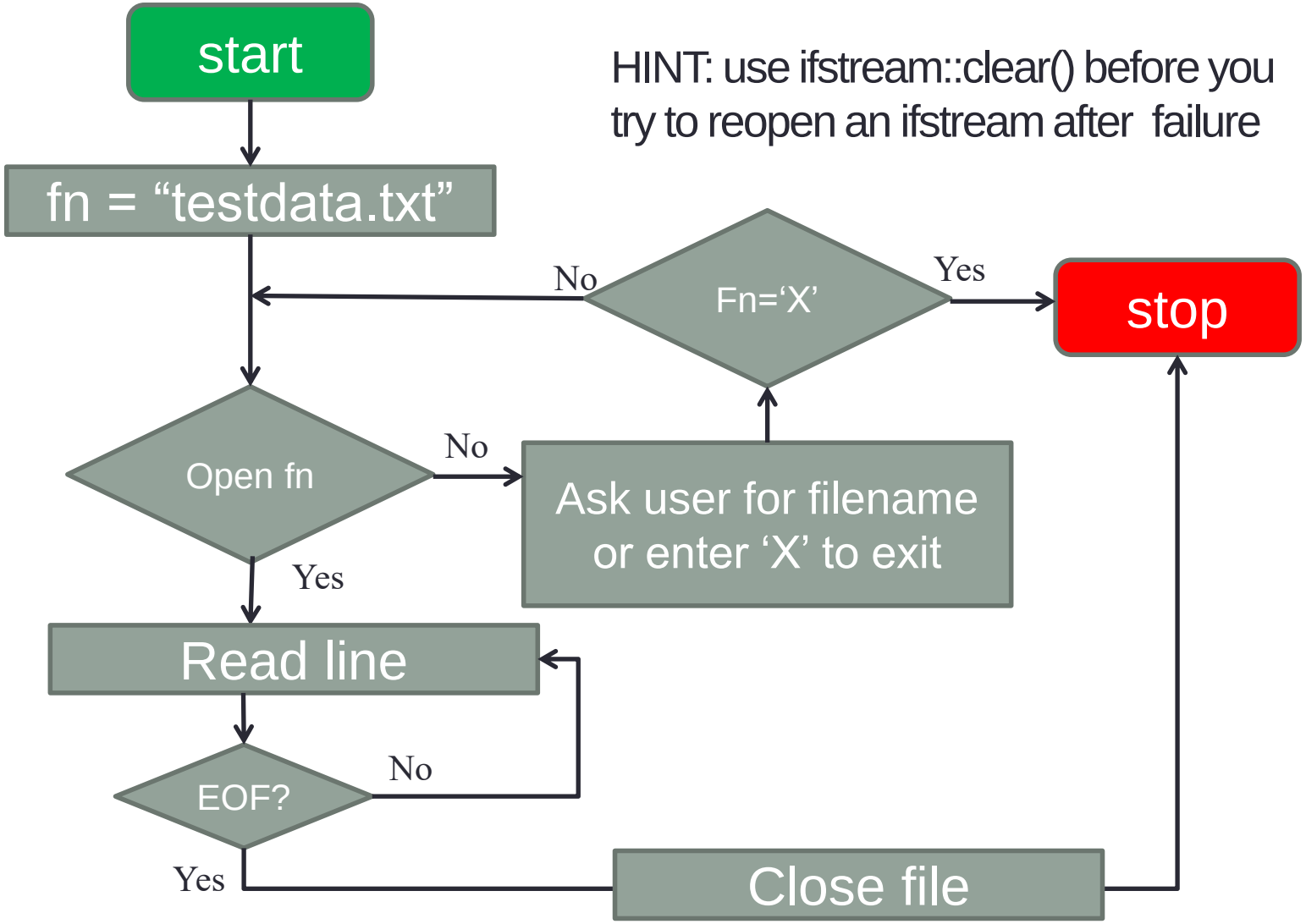
File I/O writing a file

```
std::string myfile = "MyTestFile.txt";

void writeFile() {
    ofstream myOutputfile;
    myOutputfile.open(myfile.c_str());  //could open with  flags myfile.open(MYFILE, ios::out)
                                        //note the .c_str() call on MYFILE
    myOutputfile << "Writing this to a file.\n";
    myOutputfile.close();
}
```

See http://www.cplusplus.com/doc/tutorial/files/

Please implement this in C++



HINT: use ifstream::clear() before you try to reopen an ifstream after failure

```
//============================================================================
// Name        : 3_File_IO.cpp
// Author      :
// Version     :
// Copyright   : Your copyright notice
// Description : Hello World in C++, Ansi-style
//============================================================================

#include <iostream>
#include <fstream>
#include <string>

using namespace std;
std::string MYFILE= "MyTestFile.txt";         //global to this file
const int FAIL = -1;
const int USER_CHOSE_TO_EXIT= -2;

const int SUCCESS =0;

//***README***  use parts of the following code to implement the flowchart found in
//4- Headers, Streams, Namespaces lecture

int classExercise(string &filename){
                        //code to read a file
                        ifstream myInputfile;
                        myInputfile.open(filename.c_str(), ios::in);

                        while (!myInputfile.is_open()){
                                        cout << "Please enter a filename or \'x\' to exit"<<endl;
                                        cin >> filename;
                                        if (filename == "x" || filename == "X")
                                                        return USER_CHOSE_TO_EXIT;
                                        myInputfile.open(filename.c_str(), ios::in);
                        }

                        //read and count the data
                        if (myInputfile.is_open()){
                                        std::string line;

                                        //read entire file
                                        while (!myInputfile.eof()) {                    //exits when reach end of file
                                                        getline(myInputfile, line);     //gets a line up to '/n' char
                                                        cout<<line;
                                        }
                                        myInputfile.close();
                        }
                        return SUCCESS;
}

int main() {
    return classExercise(MYFILE);
}
```