```
Script started on 2024-11-07 13:36:10-06:00 [TERM="xterm-256color" TTY="/dev/pts/6'
e_prevost@ares:~/Portfolio_2/Lab 1$ pwd
/home/students/e_prevost/Portfolio_2/Lab 1
e_prevost@ares:~/Portfolio_2/Lab 1$ show-code lab_1.cpp


lab_1.cpp:


     1  #include <iostream>
     2  #include <fstream>
     3  #include <string>
     4  #include <limits>
     5
     6  class Student {
     7  private:
     8      std::string name;
     9      long id;
    10      float gpa;
    11      char grade;
    12
    13  public:
    14      void read(std::istream& in) {
    15          std::getline(in, name);
    16          in >> id >> gpa >> grade;
    17          in.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
    18      }
    19
    20      void write(std::ostream& out) const {
    21          out << name << '\n' << id << '\n' << gpa << '\n' << grade << '\n';
    22      }
    23  };
    24
    25  bool openInputFile(std::ifstream& file) {
    26      std::string filename;
    27      std::cout << "Please enter the name of your data file: ";
    28      std::cin >> filename;
    29      file.open(filename, std::ios::app);
    30
    31      if (!file) {
    32          std::cout << "I'm sorry, I could not open '" << filename << "'.\n"
    33          return false;
    34      }
    35
    36      std::cout << "File '" << filename << "' opened successfully!\n\n";
    37      return true;
    38  }
    39
    40  bool openOutputFile(std::ofstream& file) {
    41      std::string filename;
    42      std::cout << "Please enter the name of the copy file: ";
    43      std::cin >> filename;
    44      file.open(filename, std::ios::app);
    45
    46      if (!file) {
    47          std::cout << "I'm sorry, I could not open '" << filename << "'.\n"
    48          return false;
    49      }
    50
    51      std::cout << "File '" << filename << "' opened successfully!\n\n";
    52      return true;
    53  }
    54
    55  int main() {
    56      std::cout << "\nWelcome to your worst nightmare (copying program)\n\n"
    57
    58      std::ifstream inFile;
    59      while (!openInputFile(inFile)) {}
    60
    61      std::ofstream outFile;
    62      while (!openOutputFile(outFile)) {}
    63
    64      std::cout << "Copying data...\n\n";
    65
    66      Student student;
    67      inFile.peek();
    68      while (!inFile.eof()) {
    69          student.read(inFile);
    70          student.write(outFile);
    71          inFile.peek();
    72      }
    73
    74      inFile.close();
    75      outFile.close();
    76
    77      std::cout << "you copied the data good job \n\n";
    78      std::cout << "bye bye\n";
    79
    80      return 0;
    81  }
e_prevost@ares:~/Portfolio_2/Lab 1$ CPP lab_1
lab_1.cpp***
lab_1.cpp: In constructor
‘Student::Student()’:
lab_1.cpp:6:7: warning:
‘Student::name’ should be initialized in the
member initialization list [-Weffc++]
    6 | class Student {
      |       ^~~~~~~
lab_1.cpp:6:7: warning:
‘Student::id’ should be initialized in the
member initialization list [-Weffc++]
lab_1.cpp:6:7: warning:
‘Student::gpa’ should be initialized in the
member initialization list [-Weffc++]
lab_1.cpp:6:7: warning:
‘Student::grade’ should be initialized in
the member initialization list [-Weffc++]
```

```
lab_1.cpp: In function ‘int main()’:
lab_1.cpp:66:13: note: synthesized
method ‘Student::Student()’ first required here
   66 |     Student student;
      |             ^~~~~~~


e_prevost@ares:~/Portfolio_2/Lab 1$ ls
lab_1.cpp  lab_1.info  lab_1_input_file.txt  lab_1.out  lab_1.tpq  typescript
e_prevost@ares:~/Portfolio_2/Lab 1$ ./

Welcome to your worst nightmare (copying program)

Please enter the name of your data file: lab_1_input_file.txt
File 'lab_1_input_file.txt' opened successfully!

Please enter the name of the copy file: copy_file
File 'copy_file' opened successfully!

Copying data...

you copied the data good job

bye bye
e_prevost@ares:~/Portfolio_2/Lab 1$ ls
copy_file  lab_1.cpp  lab_1.info  lab_1_input_file.txt  lab_1.out  lab_1.tpq  type:
e_prevost@ares:~/Portfolio_2/Lab 1$ cat copy_file
Jason James

123456
9.2
B
Tammy James

123457
11.2
A
Familiar Kensei James

123458
5.6
D
Quincy 2005 is Awesome

110121
8.4
B
e_prevost@ares:~/Portfolio_2/Lab 1$ cat lab_1_input_file.txt
Jason James

123456

9.2
```

```
B

Tammy James

123457 11.2 A

Familiar Kensei James

123458

5.6 D

Quincy 2005 is Awesome

110121 8.4

Be_prevost@ares:~/Portfolio_2/Lab 1$ cat la_1.info
****************

Robert Prevost

CSC 122 W01



Copy Many Lab



Takes in input and runs through library function to recursively validate

inputs until a correct input is met.



Base Level: Level 1

Total Level: Level 1

****************e_prevost@ares:~/Portfolio_2/Lab 1$ cat lab_1.tpq
1. What weird behavior does open exhibit for output files by default? How do

we fix this problem?



the open command opens the file in truncate mode which can delete data to

prevent this we set the app flag on. This specifies the file open type to

append information.
```

2. How much does spacing matter in the input file? The output file? (Hint: Would it matter if it weren't present at all? If there were many, many spaces?)

the spacing does not have a large effect on the copy function as the cin reads over whitespace as a way to indicate when the variable ends. To combat this in line we use the getline function which grabs the whole line.

3. Problems with the (C)string piece of data:

I. What problem might you have with the (C)string data (being as it is 'mixed' with so many other data types in this file: numbers and characters and such)? (Hint: Is the (C)string data one or multiple words?) Is this difficult to fix? What assumption did you make to solve this problem?

the string can be any sort of length with any white space. This could lead to issues if we used the cin approach of grabbing the variable info for all the variables. Instead, we must use the getline function for the string and the regular cin >> for the rest of the variables.

II. If the (C)string had to beplaced afterthe other data – at the end of the data group/block, what problem might arise? How do we typically avoid this situation (again, assuming the data has to be in that order)? [Assume you have re-written your code to deal with the new data order – but do not do so.]

Now this leads a different problem because we do not know how long the name might be and we need some sort of separator between Class objects. If we make sure our data is sorted so that the name always ends with a line-ending \n, then we can use the getline function to accurately grab the value of the string.

III. Think about, but do not fix, the potential problem of the user's (C)string being longer than you had anticipated. (Answer this question even if you used the string class to code your program!)

There might be formatting issues that come with reading fairly large strings. If we assume a max length for the name string an extremely large string could also cause problems.

4. I. What function is used to tell when you've reached the end of a stream?

eof() is the function we use.

II. Can this function be used on the keyboard stream?

yes but it is not practical by any means. The EOF needs to be triggered which is tough since it is not normal that a user would know what keys would need to be pressed to indicate an eof signal.

5. How do you pass a stream to a function?

```
you pass it like any other variable just make sure it is by reference and

its correct stream type.


6. Why is it a good idea to make input functions ignorant of whether or

not a stream is cin or a file? Output functions/cout/file?


They both work under the same mechanism, reading text input to append to a

variable or any other format. Therefore, why not combine them so we can use

them together? maybe some parts we want from a file and some parts we want

from user input, having it be together allows for that.


7. Why do we close files?


to ensure no data gets corrupted and free up program space if we are still

running the program. For example, I ran some code once that analyzed over

40,000 word documents but nowhere in my code did I ever close those word

documents so you can guess what happened to the memory on my laptop in a

short amount of time.e_prevost@ares:~/Portfolio_2/Lab 1$ exit
exit

Script done on 2024-11-07 13:39:14-06:00 [COMMAND_EXIT_CODE="0"]
```