

**CPE 101
Fall 2009
Laboratory 6**

Due Date

- **Friday October 30th, 11:59pm**
- **You must turn in your source electronically on vogon using the [handin](#) command – instructions are provided in below.**

Objectives

- To practice using an array of integers.
- To practice using strings.
- To practice writing simple file I/O.
- To practice simple problem solving.
- To develop a complete C programs, compile it, and turn it in electronically.

Resources

- This is an individual programming assignment; please work on your own code.
- You may use your instructor, peers, texts, and your own innate capabilities and resourcefulness!

Ground Rules

- None

Orientation:

This lab will involve developing a small C program to practice using arrays and strings and to practice reading data from a file. This program is a very simple program to process a student's grade information. We will be using parallel data to represent our data (next week we will learn about alternative data structures to represent data but please follow the specifications for this week). You will read in data from a file that has a student's name listed and then 3 numeric grades for that student. The program allows the user to specify which student to print out their grade data. The input file will look like this for example:

```
Alice 99 95 78
Tom 67 89 72
Sandy 54 78 89
```

You will represent the data as three different strings, one for each student's name and then three different integer arrays, one array for each student's grades.

Part 1: Develop a function that takes in a string and an integer array and prints out the data in the following format. Note that this function should include a for loop.

```
name: Alice grades: 72 95 0
```

Part 2: Start to develop your main function by declaring the necessary data.

Remember, you will represent the data as three different strings, one for each student's name and then three different integer arrays, one array for each student's grades.

Part 3: Continue to develop your main function to prompt the user for an input file

You will need to prompt the user for the name of the file, be sure to declare a string to store the file's name. You will also need to declare a file pointer (FILE *) and use fopen to open the file for reading.

Part 4: Read in the data from the file using fscanf. You may download the file from the examples directory or create your own matching the above format.

Part 5: Develop a loop in main to allow the user to search for a given student's name and print their grade records.

Be sure to use the print function you developed. See the example output for a sense of the loop prompts and searching prompts. Declare any necessary variables to complete your task.

Example run:

An example run of the complete program, might look something like this. User input is underlined for clarity:

Enter the filename: students.txt

Opening students.txt

Search for which student?: Tom

name: Tom grades: 89 89 0

Would you like to print another grade? 1=yes, 0=no:
1

Search for which student?: Alice

name: Alice grades: 72 95 0

Would you like to print another grade? 1=yes, 0=no:
1

Search for which student?: Andy

Name not found!

Would you like to print another grade? 1=yes, 0=no:
1

Search for which student?: Sandy

name: Sandy grades: 54 78 0

Would you like to print another grade? 1=yes, 0=no:
0

Complete the lab by handing in Your Source Electronically...

1. Transfer your files (lab6.c) to vogon as you've done for previous labs
2. Log on to vogon using the Secure Shell Client program (or your favorite equivalent).

3. Change directory (cd-command) to the directory containing the source file or files to hand in.
4. Execute the following command:

```
handin zwood csc101lab06 lab6.c
```

5. You should see messages that indicate handin occurred without error. You can (and should) always verify what has been handed in by executing the following command:

```
handin zwood csc101lab06
```