# CSC 232 Lab 5 Spring 2018

## Due at start of next week lab class- all files are to be placed in a folder named as Lab5 in your folder on Trace Server

**Submission Requirements:**

1. **Upload your source code to trace.**
2. **Copy your folder name on blackboard.**

Part 1: Introduction to Files

Open program files.cpp from the Lab 5 folder.

Exercise 1: Assume that the data file has hours, payRate, stateTax, and fedTax on one line for each employee. stateTax and fedTax are given as decimals (5% would be .05). Complete this program by filling in the code (places in bold).

Exercise 2: Run the program. Note: the data file does not exist so you should get the error message:

Error opening file.

It may not exist where indicated.

Exercise 3: Create a data file with the following information:

40 15.00 .05 .12

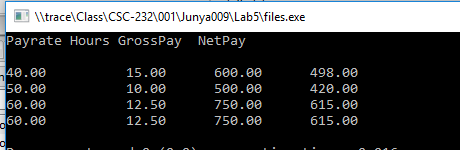
50 10 .05 .11

60 12.50 .05 .13

Save it in the same folder as the files.cpp file. What should the data file name be

payroll.dat

Exercise 4: Run the program. Record the output here:



Exercise 5: Change the program so that the output goes to an output file called pay.out and run the program. You can use any logical internal name you wish for the output file.

Part 2: Files as Parameters and Character Data

Open program Grades.cpp and the data file graderoll.dat from the Lab 5 folder.

Exercise 1: Complete the program by filling in the code (areas in bold). This problem requires that you study very carefully the code and the data file already written to prepare you to complete the program.

Exercise 2: Add another field called letter to the record which is a character that holds the letter grade of the student. This is based on the average of the grades as follows: test1 and test2 are each worth 30% of the grade while final is worth 40% of the grade. The letter grade is based on a 10 point spread. The code will have to be expanded to find the average.

90–100 A

80–89 B

70–79 C

60–69 D

0–59 F

Part 3: Binary Files and the write Function

Open program budget.cpp from the Lab 5 folder. The code is as follows:

Exercise 1: This program reads in a record with fields name, income, rent, food, utilities, and miscell from the keyboard. The program computes the net (income minus the other fields) and stores this in the net field. The entire record is then written to a binary file (indata). This file is then closed and reopened as an input file. Fill in the code as indicated by the comments in bold.

Sample Run:



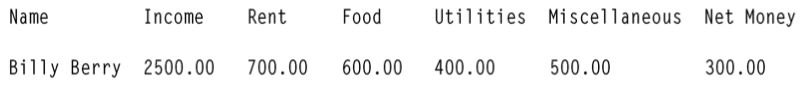
 

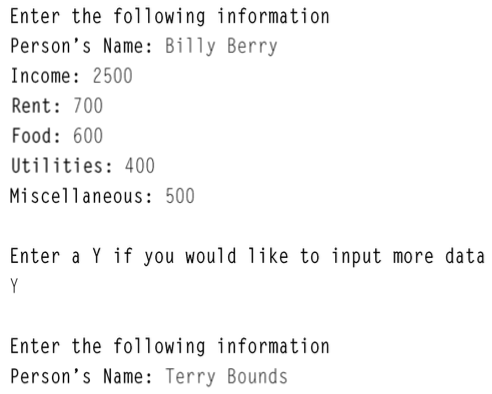
 

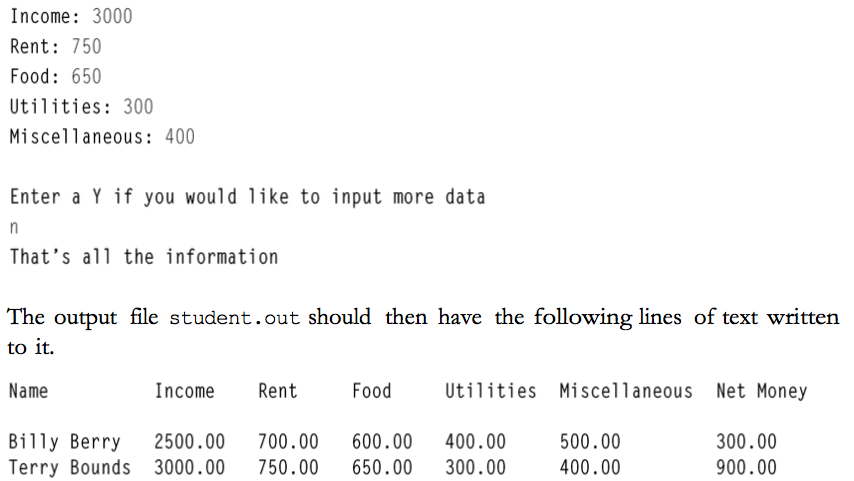
The program should write the following text lines to the output file student.out.



Exercise 2: Alter the program to include more than one record as input. Use an array of records.

Sample Run:





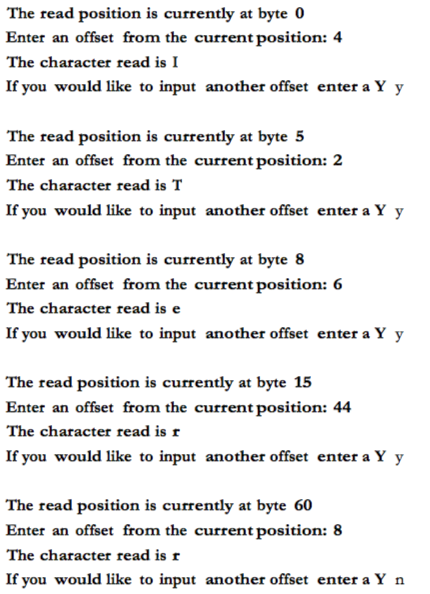
Part 4:Random Access Files

Open program randomAccess.cpp and the data file proverb.txt from the Lab 5 folder.

Exercise 1: Fill in the code as indicated by the comments in bold. The file proverb.txt contains the following information:

Now Is The Time fOr All GoOd Men to come to the aid of their 044Family

Sample Run:



Exercise 2: Change the program so that the read position is calculated from the end of the file. What type of offsets would you need to enter to get characters from the proverb? Do several sample runs with different numbers to test your program.

I made a another file named by randomAccess2\_ Exercise 2.cpp to answer this question.

* InFile.seekg(0L,ios::end) //set the read position to the end of the file;
* The offset should be an negative long;

Part 5: Write a program by yourself

Bring in the file employee.in from Lab 5 folder. Write a

This program that will read records from this file and store them in a binary file. That file will then be used as input to create an output file of the information. The data file contains employee information consisting of name, social security, department ID, years employed, and salary. In addition to displaying the information of each record, the program will also calculate the average salary and years employed of all the records. This additional information is stored in the same output file.

