Texas Tech University Whitacre College of Engineering

CS 2413/5401 – Data Structures
Spring 2022
Lab Assignment 1

Acknowledge your collaborators or source of solutions, if any. **Online submission is required.**

While designing your programs or answering items, you are free to come up with your own assumptions based upon concepts and material learned in the course, if every potential specification is not given to you. Just be reasonable and document your assumptions. Such assumptions should not conflict with concepts and material learned in the course.

Your compliance with the "PROGRAMMING STYLE GUIDELINE" for CS 2413/5401 will affect your actual grade. All assignments will be checked for academic misconduct (cheating, plagiarism, collusion, falsifying academic records, misrepresenting facts, violations of published professional ethics/standards, and any act or attempted act designed to give unfair academic advantage to oneself or another student) defined by "OP 34.12: Grading Procedures, Including Academic Integrity" of TTU. Special software will be used to uncover such attempts.

A subset of answers submitted in this lab may be graded.

Objective: Practice concepts of C programming, complexity analysis, and abstract data types

Tasks:

1. For the code below, please do the following:
   - **a.** Put item answers and sample code runs in comments at the end of the submitted code file.
   - **b.** Type in the code making it look better stylistically, compile the code, and run the code showing the input and output. What does the code do?
   - **c.** What would happen if the *'s were deleted from `fun1`'s parameter list and the &'s were deleted off the arguments in the calls to `fun1`? Make those changes in the code and run the code. If it differs from the original code, why?

```
#include <stdio.h>
void fun1(double *smp, double *lgp);
int main (void) {
double num1, num2, num3;
printf("Enter 3 numbers separated by blanks> ");
scanf("%lf%lf%lf",&num1,&num2,&num3);
fun1(&num1,&num2);
fun1(&num1,&num3);
fun1(&num2,&num3);
printf("The numbers are:  %.2f  %.2f  %.2f\n", num1, num2, num3);
return (0);
}
void fun1 (double *smp, double *lgp) {
double temp;
if (*smp > *lgp) {
temp=*smp;
*smp=*lgp;
*lgp=temp;
}
}
```

2. In this problem, use the least squares technique to determine a linear model (y = mx + b) for estimating the ozone-mixing ratio at a specified altitude.
   a. For this problem, please put item answers and sample code runs in comments at the end of the submitted code file.
   b. Create a file "zone.txt" in the same folder as the least squares C program with the following values in altitude (km) and ozone mixing ratio (ppmv).
      i. 20 3
      ii. 24 4
      iii. 26 5
      iv. 28 6
   c. ***C file usage help for those who have not had C before***
      i. Include stdio.h
      ii. Declare a file pointer variable: FILE *infile;
      iii. Open the file and use an if statement to make sure infile is not NULL:
         1. infile = fopen("zone.txt","r");
      iv. Read from the file
         1. fscanf(infile,"%lf %lf",&x,&y);
         2. fscanf returns the number of values input successfully.
      v. Close the file: fclose(infile);
   d. Create a C program to input the file, calculate the slope, m, and the intercept, b, using the following equations, and output m and b.
      i. $\boldsymbol{m = \frac{\sum_{k=1}^{n} x_k * \sum_{k=1}^{n} y_k - n * \sum_{k=1}^{n} x_k y_k}{(\sum_{k=1}^{n} x_k)^2 - n * \sum_{k=1}^{n} x_k^2}}$
      ii. $\boldsymbol{b = \frac{\sum_{k=1}^{n} x_k * \sum_{k=1}^{n} x_k y_k - \sum_{k=1}^{n} x_k^2 * \sum_{k=1}^{n} y_k}{(\sum_{k=1}^{n} x_k)^2 - n * \sum_{k=1}^{n} x_k^2}}$
   e. For example, with the file zone.txt,
      i. $\boldsymbol{\sum_{k=1}^{4} x_k = 20 + 24 + 26 + 28 = 98}$
      ii. $\boldsymbol{\sum_{k=1}^{4} y_k = 3 + 4 + 5 + 6 = 18}$
      iii. $\boldsymbol{\sum_{k=1}^{4} x_k^2 = (20)^2 + (24)^2 + (26)^2 + (28)^2 = 2436}$
      iv. $\boldsymbol{\sum_{k=1}^{4} x_k y_k = 20 * 3 + 24 * 4 + 26 * 5 + 28 * 6 = 454}$
      v. $\boldsymbol{m = \frac{(98*18 - 4*454)}{(98)^2 - 4*2436} = 0.37}$
      vi. $\boldsymbol{b = \frac{(98*454 - 2436*18)}{(98)^2 - 4*2436} = -4.6}$
      vii. Example output:
         1. Range of altitudes in km: 20.00 to 28.00
         2. Linear Model: ozone-mix-ratio = 0.37 * altitude + -4.60
   f. When given a problem like the above, we would like the code to be efficient in space and time. For example, are there shortcuts that can be identified in the problem specification, do we have to use an array, is there another data structure that would offer an improvement in time or space complexity, or are multiple nested loops needed, etc.
      i. Space complexity
         1. How much memory is needed to store the data?
         2. Is an array needed?
      ii. Time complexity
         1. What is f(n) for this program?
         2. Show f(n) <= O(g(n))
         3. For example, if the program is f(n) = $3n^2$, then show $3n^2$ is $O(n^2)$ similarly to slide 29 in the chapter 2 lecture notes on Blackboard.

3. Give the data structure that would be most helpful for writing a C program to do the following:
    a. Sorting n integers
    b. Scheduling jobs to a CPU in order of submission
    c. Storing recursive function calls (each time a function is called, an activation record is created; this record is deleted when the function returns)
    d. Storing the folder structure of a user account
    e. Showing the friend relationships among social media users

Learning Outcomes:

- Understand C program concepts, such as variables, operator precedence, selection, looping, functions, pointers
- Begin to understand how to analyze the complexity of a program or algorithm
- Begin to understand abstract data types

Grading: 45 points

Standard Deductions - 10 points
Problem 1 – 10 points (problem1.c)
Problem 2 – 20 points (problem2.c, zone.txt)
Problem 3 – 5 points (problem3answers.txt)

Due Date:

Friday 1/21/2022, 11:59pm (submitted on Blackboard)