

ASSIGNMENT FIVE

EVERYTHING INCLUDING STRUCTS, TYPEDEF

CS2263, Fall 2021

OUTCOME

Develop and test a program that assigns students to school bus routes

BACKGROUND

In manipulating/editing geographic data, it is often necessary to identify proximity to a feature. For example, each fall it could be necessary to assign students to school bus routes, based on the student's home location and the school bus stops for each bus route. Each student's bus route could be determined as the bus route with the bus stop closest to the student's house.

YOUR TASK

Write a C program to assign students to bus routes by selecting a particular bus route (set of bus stops) from a set bus routes. A bus route is an array of Point2D types with a route name (String). Bus routes are kept in a file and will need to be read in and held as an array of bus routes. A student will be represented with their location (Point2D) and name (String). The program will read in a file (name specified as a command line argument) of bus route information and store it. The program will then process students, read from standard in (no prompts), determine which route and stop is closest, and report the student and the route name and stop number.

Your program should use a module for lines (array of points and string) and this module should have the ability to determine the route/stop for the given student location (point). If you're feeling ambitious, you could even build a LineList module that manages a set of lines.

```
$ cat <student-file> | busAssignment <busroute-file>
```

You should use all appropriate modules that you have previously built for the course. All file reading should be done using functions within the appropriate module.

STUDENT RECORD STRUCTURE

Each student is represented as the coordinate location followed by a space and their name. Each student is held on a separate line:

```
23.44 55.2 Ricky Wightman
```

BUS ROUTE FILE/RECORD STRUCTURE

The bus route file's first line is an integer representing the number of bus routes in the file. Each record (line) after this holds an integer representing the number of bus stops (points) followed by the coordinates and ending with the name of the bus route:

```
2 0 0 1 1 Bailey Drive
```

Use the provided data to test your program. File structures cannot be altered – this format is what will be used to test your program.

NOTE

To use this file format you will need to be able to dynamically allocate arrays, like you can in Java. This requires the `-std=c99` flag in your compile command.

Your program will be marked against a different set of files.

THE REPORT

In a few sentences describe each module that you design for your program. Focus on what each of the data structures holds and how each of the functions acts on them.

Show the testing of one of the functions using a test program.

Show the output from running your program on the included test data.

SUBMISSION FORMAT

- Before the due date for this assignment, students should submit a single zip or tar file (named *LastName_FirstName_A5.zip* or *LastName_FirstName_A5.tar*) online to the lms containing:
- Your report as a pdf file
- Your source code directory:
 - This should include all of your source files, including test programs and makefile.
 - This should not include object (.o) files and executables. Nobody needs to see those.