Programming Assignment #5 -Binary Trees and Recursion (50 pts)

This assignment is similar to #4 except

- The Course information should be be represented as an ascending ordered binary tree. (Keep the tree nodes in order as they are inserted.)
- Your search for a courseID is using a recursive binary tree traversal.
- The command file will have a new subcommand, TPRINT, for COURSE. We will be able to print courses in a tree format.
- You will be provided with a driver program (cs1713p5Driver.c) (see below)
- Your code must be created in a separate C file (cs1713p5.c). (see below)
- There is a new include file, cs1713p5.h
- Several of the functions must be recursive: searchTree, insertTree.
- Consider using a makefile to reduce the chances of errors typing in your gcc comands.

Input:

Course — same as Programming Assignment #4; however, instead of placing it in linked list, you will put it into an ascending ordered binary tree.

Command — Same as assignment 4 plus this new subcommand for COURSE:

`COURSE TPRINT`

This prints the binary tree in a structured format. The code for this print is provided in the driver, your code will simply call printFormattedTree.

Driver program:

You will be provided with a driver program, cs1713p5Driver.c which

1. invokes the driver's processCommandSwitches
2. invokes the driver's getCourses to read the original course information into a binary tree using your insertTree. You must keep the tree ordered after each insertion.
3. invokes your printCourses to print the original course information.
4. invokes a driver-provided processCommands which
    - reads input lines from the command file until eof:
        - prints the input line
        - determines command and subcommand
        - invokes either
            - your processStudentCommand to process a STUDENT
            - your processCourseCommand to process a COURSE subcommand
5. invokes your printCourses to print the resulting course information in order

Your cs1713p5.c code:

- You should probably copy your cs1713p4.c into a file named cs1713p5.c.
- It must **not** include `cs1713p5Driver.c` within your cs1713p5.c file.
- You must create the following routines (see the include file):
    - printCourses - prints the course information (most of this is in program #4) this will be an in-order traversal of the tree. This is not the same as printFormattedTree
      processStudentCommand - processes the various STUDENT commands (most of this is in program #4)
    - processCourseCommand - processes the various COURSE commands including the new COURSE TPRINT.
    - searchTree - searches a binary tree recursively to find a course ID.
    - insertTree - inserts a course into a binary tree using the reconstruct approach.

Please review the cs1713p5.h include file.

Compiling using the **make** utility:

- You might want to use your make file from the previous project.

Executing the p5 executable:

```
./register -c p5Command.txt -s p5Course.txt
```

Turn in:

Your include file (if it changed)
Your cs1713p5.c
Your output based on the data provided.
Your modified makefile