CIS 26B
Advanced C
Programming Assignments

Homework 5

Binary Search Trees and Queues

100 Points

Projects

`26B_H_5A` – Multiple Source and Header Files in C Programs

- Read: `header_files.pdf`
- Build and run the project consisting of three source files and one header file. Update the project and run it again. The updating instructions are given as comments.

`26B_H_5B` – Cross-Reference List (details below and on the next pages)

Grading

Program 5A – 25 Points
Program 5B – 65 Points

1. Timestamp – 5
2. BST – 20
3. Queue – 10
4. Other functions – 15
5. 2 or 3 source files, 1 or two header files – 10

Memory Management – 5 // No memory leakSelf-
Assessment Report: – 5Points

Write a short report (`26B_H5Report.docx`), briefly explaining your code and containing an assessment of your implementation based on the above grading criteria

Cross-Reference List

Write a program which generates a cross-reference listing of the identifiers in an input C program. A cross-reference file will list identifiers in alphabetical order and their line numbers like this:

```
boat          3    4
row           2    3    4
your          3    4
```

Identifier `boat` is found on lines 3 and 4, `row` is found on lines 2, 3, and 4, and `your` is found on lines 3 and 4.

CIS 26B
Advanced C
Programming Assignments

The input and output files will be specified on the command line. Make sure the input file is a C program i.e., name ends with ".c".

Your cross-reference listing will consist of an ASCII-ordered listing of identifiers in the input program plus the line numbers where those identifiers occur. Make sure your cross-reference listing is nicely formatted and timestamped. Make line numbers line up under one another.

Put the identifiers in a tree and the line numbers for a given identifier in a queue which hangs off of that identifier's tree node. No duplicate line numbers.

Note that you assume a correct C program. A C identifier starts with a letter or underscore followed by any combination of letters, underscores, and digits. No identifier occurs in comments or quoted strings (single or double).

There is no interactive part in this program: the program builds a binary search tree with a queue in each node, then creates the cross-reference list by writing the BST to a file. The only thing to be displayed to the screen is a title, a brief description of the program, and the names of the input and output files.

Example
Assume the input file contains the following text (not a C program, but good for testing):

```
row, row,
row your boat
Row, row your boat.
```

The output file will contain a title, the original text with a line number added, the timestamp, and the cross-reference list:

```
~*~ Cross-Reference List ~*~

    1|
    2| row, row,
    3| row your boat
    4| Row, row your boat.
    5|

Cross-reference list made at: Mon Mar 15 17:19:30 2021

Row        4
boat       3   4
row        2   3   4
your       3   4
```

CIS 26B
Advanced C
Programming Assignments

Header and Source Files

Design A:

- Create one source file for basic BST and queue functions
- Create one header file for basic BST and queue functions
- Create one source file for main() and other functions

Design B:

- Create one source file for BST functions definitions
- Create one header file for BST functions declarations
- Create one source file for queue functions definitions
- Create one header file for queue functions declarations
- Create one source file for main() and other functions

Design C:

- Create one source file for basic BST and queue functions
- Create one source file for main() and other functions
- Create one header file for basic BST and queue functions, and other functions (like in Program 5A)

Include only the functions you need in this project. For instance, we are not deleting or searching the BST, therefore you do not have to implement these functions.

Testing

Run your program twice using the following input files:

`song.c`
`test.c`