

In-Lab Question

Title: Department Information Processor_2

Objective:

Develop a C program to read department names from a given file “**input.txt**”, But due to some virus the contents were reorganized as following:

- The first contains an integer n indicating number of departments in the database
- The second line defines q , the number of queries which need to be processed.
- The next $n/2$ lines indicating the first $n/2$ departments of the database
- The next q lines indicates the queries which need to be processed
- The remaining lines having the remaining department names in database

Part -1

Now for each query do the same operation as done in the lab 1 regarding the department name based on whether it is found in the file or not

Requirements:

1. Input:

- You are given an input file “input.txt”
- Process the input as explained earlier in the input file description

- Note while parsing the input file strings shouldn't contain "\n". For example "mech" should not be read as "mech\n"

2. Search Functionality:

- Implement a function to search for the user-provided department name in the token array.
- If the department is found, store its index in the id field of the department struct.
- If the department is not found, set the id field to -1.

3. Name Transformation:

- If the department is found, transform the name by converting the first letter to lowercase and the remaining letters to uppercase.
- If the department is not found, reverse the first half of the user-provided department name.

6. Output:

- Print the original department name, the department ID (or indicate that it was not found), and the transformed department name in an output file "output.txt".

Test Cases for the Given Input File

Sample input file:

6

5

chem

mech

computers

computers

arts
mech
drama
plane
bio
chemical
maths

Sample output file:

Query number 0:

Department name: computers

Department ID: 2

Transformed name: cCOMPUTERS

Query number 1:

Department name: arts

Department ID: -1

Transformed name: rats

Query number 2:

Department name: mech

Department ID: 1

Transformed name: mECH

Query number 3:

Department name: drama

Department ID: -1

Transformed name: rdama

Query number 4:

Department name: plane

Department ID: -1

Transformed name: lpane

Part-2

Transform this into three source files and a Makefile:

main.c: Move the main() function here. Ensure it includes necessary headers and uses the functions defined in department.c and misc.c.

department.c: Implement the search_department(), letter_transform(), and reverse_first_half() functions.

misc.c: Implement all utility functions in this file

department.h: Declare the department structure and prototypes for functions in department.c.

misc.h: Declare prototypes for functions in misc.c and include any necessary macros.

Create a Makefile to compile main.c, department.c, and misc.c into an executable named str_read. Ensure the Makefile includes targets for building and cleaning up the project.

