Introduction to Computers and Programming

Homework 8

2023/11/14

1. Deadline

You have one week to complete the homework. Hand in your homework via E3 before 2023/11/21 23:55. Please finish your homework as soon as possible. In addition, make sure that your code can be executed on Visual Studio Community 2019.

2. Problems

2.1 Library management(100 points)

In this task, you are going to manage 2 sets of books in the library.
Use 2 arrays **A** and **B** to store book titles in the library. Each element of the array holds one book title. You need to write a program that allows for adding, deleting, or exchanging elements of **A** and **B** repeatedly.
First, read 2 positive integers, **m** and **n**. Then dynamically allocate **A** with size of **m**, and **B** with size of **n**. Each element is a pointer to a char array.
Second, read an integer **p** (p=0, 1, 2 or 3) to choose your operation.

- **p**=0 is for adding a book:
  - Read 3 integers **x**, **y**, and **s**:
    - **x** should be 0 or 1 to choose array **A** or **B**.
    - **y** is the index of the array and it should be empty, otherwise stop this operation. If **y** is out of bound should stop too.
    - **s** is the size of your book title ready to input.
  - Dynamically allocate a char array with size **s**, and input the string. Then store the string to the chosen index of the chosen array.
- **p**=1 is for deleting a book.
  - Read 2 integers **x** and **y**.
    - **x** should be 0 or 1 to choose array **A** or **B**.
    - **y** is the index of the array. If **y** is out of bound, stop this operation.
  - Delete the string in the chosen index of the chosen array.
- **p**=2 is for exchanging a book.
  - Read 2 integers **x** and **y**.
    - **x** is the index of the array **A**.
    - **y** is the index of the array **B**.
    - Either **x** or **y** is out of bound should stop this operation.
  - Exchange 2 chosen strings.
- **p**=3 is for quitting the process.

Third, print out every book titles in array **A** and **B**.
Repeat second step.

Input

```
m n          (0 ≤ m, n ≤ INT_MAX)
```

Repeat:

```
    p            (p = 0, 1, 2 or 3)
    x y[ s]      (x = 0, 1, or ≤ m; 0 ≤ y ≤ m or n; 0 ≤ s ≤ INT_MAX)
```

Output

`A:`

Every book titles in array **A**.

`B:`

Every book titles in array **B**.

Example:

Input

```
3 4

0
0 0 3
abc

0
1 2 5
pgrst

2
0 2

1
0 0

3
```

```
A:
abc
(null)
(null)
B:
(null)
(null)
(null)
(null)

A:
abc
(null)
(null)
B:
(null)
(null)
pgrst
(null)

A:
pgrst
(null)
(null)
B:
(null)
(null)
abc
(null)

A:
(null)
(null)
(null)
B:
(null)
(null)
abc
(null)
```

Output

2.2 String Functions(extra 20 points)

In this task, you are going to write the functions related to string by yourself.
There are 7 functions in total.

char *mystrchr(const char *s, int c);
char *mystrrchr(const char *s, int c);
size_t mystrspn(const char *s, const char *accept);
size_t mystrcspn(const char *s, const char *reject);
char *mystrpbrk(const char *s, const char *accept);
char *mystrstr(const char *haystack , const char *needle);
char *mystrtok(char *str, const char *delim);

All the above functions must have the same functionality as the standard functions in C. (You can check what these functions do in C from the Internet.)
Download the sample code from E3, and complete TODO parts.
If you use these standard functions in C, you will get 0 score.

Example:

There is no example input and output in this task, you need to test whether the functions work properly yourself.

3. Submission format

Your submission should follow the format below, or you might get some penalty for the wrong format.

- xxxxxxxxx_hw_w10.zip
  - xxxxxxxxx_hw_01.cpp
  - xxxxxxxxx_hw_02.cpp

xxxxxxxxx is your student ID