

Lab 6 D

Problem 1: The Secret Book Club

You're the manager of a secret book club where each book is passed from one member to the next in a specific order. One day, you receive a request to find and remove a specific book from the chain. Your job is to locate the book with the mistake and remove it, ensuring the flow continues uninterrupted. The member before the book must pass the next one in line smoothly. If the requested book isn't in the club's collection, you leave the list unchanged.

Functions to Implement:

- `struct Node* deleteByValue(struct Node* head, int x);`
- `struct Node* newNode(int data);`
- `struct Node* appendNode(struct Node* head, int data);`

Input Format:

- The first line contains an integer `n`, representing the number of books in the club.
- The second line contains `n` integers representing the book IDs in the order they are passed from one member to the next.
- The third line contains an integer `value`, representing the book ID that needs to be removed.

Constraints:

- $1 \leq n \leq 1000$
- $1 \leq id \leq 10^6$

Output Format:

Return the updated list of book IDs after removing the book with ID `value`. If `value` is not found, return the original list.

Input 1:

```
5
1 2 3 4 5
3
```

Output 1:

```
1 2 4 5
```

Input 2:

```
5
1 2 3 4 5
10
```

Output 2:

```
1 2 3 4 5
```

Explanation:

In the first example, the book with ID 3 is removed, and the remaining books are passed smoothly: 1 2 4 5.

In the second example, since the book with ID 10 is not in the club, the list remains unchanged: 1 2 3 4 5.

Important Notes:

- Input and output handling is already provided. You only need to implement the required functions.
- Solutions that do not use a linked list will result in a score of 0.

Problem 2: Sum of Two Numbers Represented by Linked Lists

You are given two non-negative integers represented as linked lists, where each node contains a single digit. Your task is to write a function that adds these two numbers and returns the sum as a linked list.

Approach:

- Traverse both linked lists, digit by digit, and compute their sum.

- If the sum of the digits exceeds 9, carry over the extra value to the next digit.
- Continue until both lists are fully traversed, and make sure to account for any remaining carry value.
- The result should also be stored as a linked list, similar to the input format.

Hint:

- Working with **reversed** linked lists are easier!

Input Format:

- First line consists of an integer n representing the size of first linked list.
- Second line consists of an n characters representing the first number.
- Third line consists of an integer m representing the size of second linked list.
- Second line consists of an m characters representing the second number.

Output Format:

The sum of the two numbers.

Constraints:

- Each linked list contains digits between 0 and 9.
- The numbers do not have leading zeroes, except for the number 0 itself.
- $1 \leq n, m \leq 100$

Example Input 1:

```
4
9340
3
721
```

Example Output 1:

```
10061
```

Example Input 2:

```
3
342
```

3
465

Example Output 2:

807

Problem 3: Reverse Subset of Linked List

Asator has recently learnt to reverse linked lists for his project from his CPRO TAs. However, he realized this is not enough for his project as he needs to reverse a specific subset of the linked list. Can you help him to do this?

Formal Statement:

Given the head of a singly linked list and two integers l and r where $1 \leq l \leq r$, reverse the nodes of the list from position l to position r , and return the modified list.

Note: This must be done in-place, without creating a new list. Creating a new linked list will result in a score of 0.

Input Format:

Each test consists of one test case. The description of the test cases follows:

- The first line of each test case contains three integers n , l , and r , representing the length of the linked list, the starting index, and the ending index.
- The second line contains n integers representing the values of the linked list.

Output Format:

Output the modified linked list.

Explanation:

- Both l and r indexes are inclusive in the reversal.
- The linked list is 1-indexed, meaning the first element of the linked list corresponds to index 1.
- Only solutions using linked lists are allowed, and solutions that use arrays directly will receive a score of 0.

Sample Test Case:

Input:

```
5 2 4
1 2 3 4 5
```

Output:

```
1 4 3 2 5
```

Constraints:

- $1 \leq n \leq 1000$
- $1 \leq l \leq r \leq n$

Submission Guidelines

Do not rename any files given in the handout. Only write the code in the specified C files in the respective directories.