Lab 8 A

Problem 1: Square of a number

Bots have taken over the world and give you a chance to survive if you solve this task. You are asked to print the square of a number. Due to some issue while providing the input, the bot has added a space in between every digit in the input number. You have to account for this and solve the task.

The input has been provided in a text file by the name "number.txt". So while reading the file, use this exact file name. There is no additional input to this task. Use the regular file input method to obtain the input and print your answer onto the terminal(stdout).

Note: You would be given the space seperated digits but the number of such digits wouldnt be told beforehand. Try experimenting with fgetc method along with EOF for file parsing

File Format:

- File contains only 1 line of input
- x space seperated digits corresponding to the number
- x will not be provided in the input

Output Format:

Print 1 line containing the square of the number.

Constraints:

- $1 \leq x \leq 9$, x corresponds to the number of digits in the number.
- Each digit ranges from 0 to 9. You are guarenteed that the first digit is not a 0 and the resulting number is a valid number.

Example File:

```
1 2 3 4
```

Example Output 1:

```
1522756
```

Example File:

```
1 0 0 0 1
```

Example Output 2:

```
100020001
```

Problem 2: Narain's Secret Linked List

Narain found a lot of binary files containing an XOR-encoded linked list. Each integer is encoded using a secret key: 123. He needs your help to decode and print the original values!

Functions to Implement:

- `NodePtr loadList(const char *filename);`
- `int xorEncodeDecode(int value);`

Input Format:

- None
- There will be a dat file that will be taken as an input which needs to be decoded. It contains a Linked List filled with numbers `N`

Constraints:

- -10e18 ≤ `N` ≤ 10e18

Output Format:

Print all the numbers in the Linked List followed by 'NULL'

Output 1:

```
10 15 20 30 NULL
```

Output 2:

```
10 11 12 13 14 NULL
```

Important Notes:

-Do look at the constraints!!! Function heading can change.
- The method on how a value is saved in a dat file is also given. Have a keen look at that.

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 `l <= 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 ≤ n ≤ 1000`
- `1 ≤ l ≤ r ≤ 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.