CS162 – Programming Techniques

Lab 04
Singly Linked List

Cảm ơn thầy Trần Duy Quang đã cung cấp template cho môn học

Department of Software Engineering-FIT-VNU-HCMUS

1

Notes

Create a single solution/folder to store your source code in a week.

Then, create a project/sub-folder to store your source code of each assignment.

The source code in an assignment should have at least 3 files:

- A header file (.h): struct definition, function prototypes/definition.
- A source file (.cpp): function implementation.
- Another source file (.cpp): named YourID_Ex01.cpp, main function. Replace 01 by id of an assignment.

Make sure your source code was built correctly. Use many test cases to check your code before submitting to Moodle.

Name of your submission, for example: **18125001_W01_07.zip**

2

Content

In this lab, we will review the following topics:

- How to init, insert (head, middle, tail), delete (head, middle, tail) a linked list?

3 Assignments

A: 1 problems / assignments.

H: 8 problems / assignments.

- **Assignment 1 – 8**

Input & Output file format: a sequences of integers, end at zero, for example: 10 20 30 40 0 (a linked list of 4 integers).

3.1 Remove all x

Load a sequence of integer numbers from a text file. Ask user to enter a value x. If x exists, remove all occurrences of x out of the list. Otherwise, do not thing. Save the list to another text file.

Input:

1 2 2 4 2 6 0

User enter 2

Ouput:

1 4 6 0

3.2 Remove duplicates

Load a sequence of integer numbers from a text file. If an element appears twice or more, remove it so that it appears only once. Save the list to another text file.

Input:

1 2 2 4 2 6 0

Ouput:

1 2 4 6 0

3.3 Reverse the list

You are given a singly linked list in which each node represents an integer number. The problem is to reverse the linked list. Please write a program to input the linked list (from a text file), reverse the linked list, and output the nodes in the new order (to another text file with the same format as your input file). Remember that you are not allowed to reverse list at the input step.

Input:

1 2 2 4 2 6 0

Output:

6 2 4 2 2 1 0

3.4 Insert even numbers

Please insert nodes numbered 2, 4, 6, 8, 10… before every nodes in the list. After that, save the list to a text file.

Input:

1 2 2 4 2 6 0

Output:

2 1 **4** 2 **6** 2 **8** 4 **10** 2 **12** 6 0

3.5 Sorted list

Given a linked list of integers sorted from smallest to largest (head to end). Insert a new integer into the linked list so that it remains sorted.

Input:

27

10 20 30 40 50 60 0

Output:

10 20 **27** 30 40 50 60 0

3.6 List of sum

Given a linked list, ***in,*** create a new linked list, ***out,*** of the same length, such the node i of the ***out*** contains the sum of the data in ***in***'s nodes up to and including node ***i*** of list ***in***.

Input:

1 2 2 4 2 6 0

Output:

1 3 5 9 11 17 0

3.7 1 list → 2 lists

Given a linked list, re-arrange its nodes into two lists: <1st node, 3rd node, 5th node…> and <2nd node, 4th node, 6th node…>. Do not allocate any new nodes.

Input:

10 20 30 40 50 0

Output:

10 30 50 0

20 40 0

3.8 2 lists → 1 list

Given two linked lists, combine their nodes so that the nodes of the new list alternate between those of the two original nodes: <1st node of 1st list, 1st node of 2nd list, 2nd node of 1st list, 2nd node of 2nd list…>. Do not allocate any new nodes.

Input:

10 30 50 70 90 110 0

20 40 60 0

Output:

10 20 30 40 50 60 70 90 110 0

3.9 join 2 lists

Given the two linked lists, headed by left and right, set the last linked of the left list to point to the right list, thus joining them into one list. Do not allocate any new nodes.

Input:

10 30 50 70 90 110 0

20 40 60 0

Output:

10 30 50 70 90 110 20 40 60 0

3.10 list → number

Given a linked list of integers from 0 to 9 (inclusive), representing a non-negative integer in decimal, compute into an unsigned variable, the integer that the list represents.

Input:

1 2 3 4 5 6 0 -1

Output:

1234560

3.11 number → list

Given a non-negative integer, create a linked list of integers between 0 and 9, representing the integer. 0 is represented by an empty list.

Input:

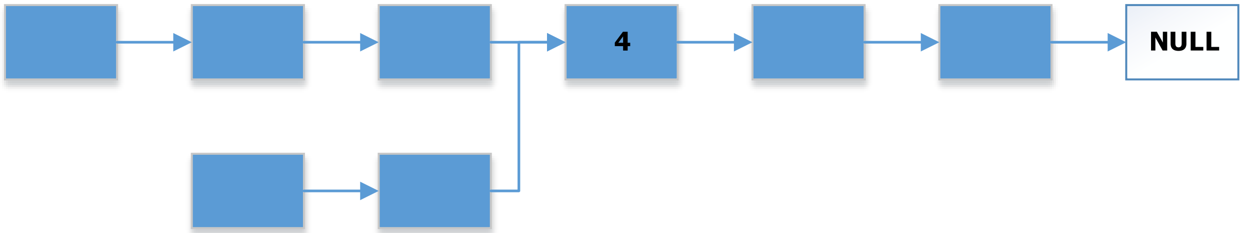
1234560

Output:

1 2 3 4 5 6 0 -1

3.12 shared node

Given 2 linked lists A and B. You need to find out if A and B have any common nodes. If so, you should find the node at which A and B are joined.



In this case, the two linked list are actually joined, and you should output (to another text file) the node 4.

Input:

1 2 3 4 5 6 0

10 20 0

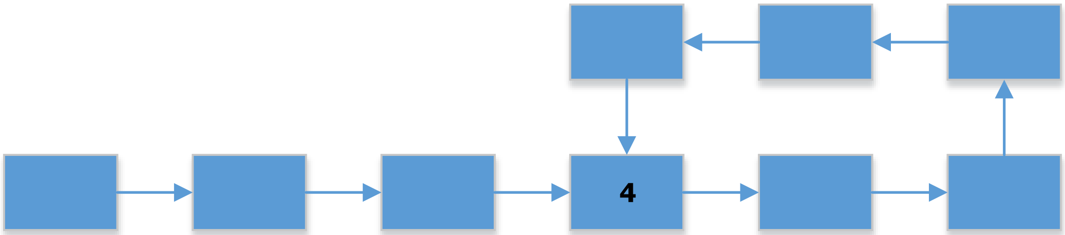
4 (if 4 exists in lst1, lst2.tail.next = node 4)

Output:

4

3.13 loop list

Given a singly linked list, you are to dertermine whether it contains a loop or not. Output YES or NO to a text file.



Input:

1 2 3 4 5 6 7 8 9 0

4 (if 4 exists in lst, lst.tail.next = node 4)

Output:

YES

3.14 Bookstore

You are asked to write a program managing a bookstore. Each book has the following information:

- Title: the title of the book (maximum 200 characters)
- ISBN: the ID of the book (10 characters)
- Author: the name of the author (maximum 40 characters)
- Language: the language of the book (maximum 30 characters)
- Year Published: the year it was published
- Price: the price of the book (in dollars).
- Stock level: in integer number representing the stock level of the book.

1. Initialization: start the bookstore with zero book.
2. Input a book with all details into the bookstore. If this book has existed in the store, update its stock level.
3. Sell a book: input an ISBN, print out the name and the price of the book. Then, reduce the number of stock level of that book. If the book is out of stock (i.e. level is zero), print out "OUT OF STOCK."
4. Find a book: input the name, print all the books (ISBN and title) whose titles contain the name as a subset.
5. Remove all book whose stock level is less than a threshold k.