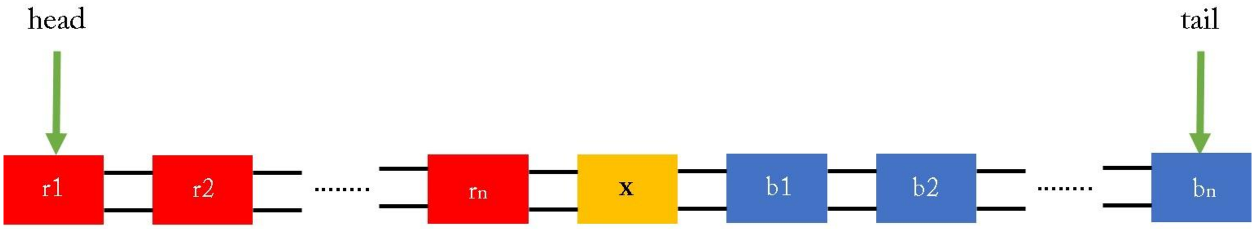
Military Institute of Science and Technology
Department of Computer Science and Engineering
Level-2, Spring Term
CSE-204 (Data Structures and Algorithms 1)
Date: 22 Apr 2021

Time: 30 mins **Full Marks: 20**

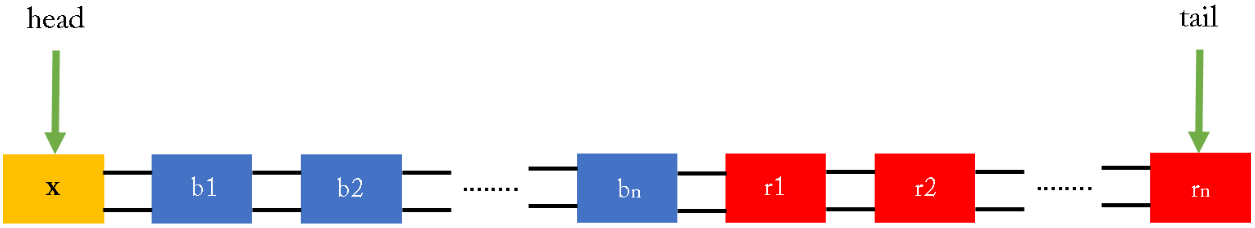
Statement:

Samee is very much uncomfortable with Doubly Linked List and so he is practicing more and more problems on it. He tries to create his own problem and then solve that. For today's practice he has colored a doubly linked list with 3 distinct colors (Red, Yellow and Blue) as the following image.



1. Yellow color represents a single node with the last occurrence of ***x*** in the doubly linked list
2. Red color represents nodes located at the left of the yellow node in the doubly linked list
3. Blue color represents nodes located at the right of the yellow node in the doubly linked list
4. Dotted lines in the figure are representing the continuation of nodes

Samee is trying to cut all the red nodes of the list sequentially and paste it after the blue nodes. So after the operation the state of the linked list will be as follows with the updated head and tail



But you know that Samee keeps chatting a lot at Messenger during the class time and so he missed some important portion of the last class. As a result he is stuck to solve this problem and asking for your help. But as you were very much attentive in the last class and also completed the assignment in time so this is not a big deal for you. To do this task you are now going to write a function: ***void cut_paste(int x)*** where ***x*** represents the value of the yellow node and taken as input from the user. You are going to use the following functions inside ***void cut_paste(int x)*** from your last assignment

1. Node* findLastOccurrence(int x) for finding the yellow node
2. void printList() for printing the list after performing all required operations

Create a separate menu with ***choice = 12*** for executing the ***cut_paste*** function.

Special Instructions:

For simplicity you can assume the following instructions:

1. At least 3 values are inserted in the list before performing the ***cut_paste*** function
2. The yellow node will never occur as head or tail before performing the ***cut_paste*** function

Input:

First insert at least 3 values in the list. Then perform the ***cut_paste*** function by taking 12 as choice from the user. Take ***x*** as input from the user where ***x*** represents the value of the yellow node.

Output:

1. Print the list after performing the desired operation if the yellow node is found in the list
2. Do nothing, just print the previous list if the yellow node is not found in the list

Sample:

Example No	State of the list before ***cut_paste***	Value of the yellow node (***x***)	State of the list after ***cut_paste***
Example-1	1 2 6 8 2 9 2 8 5	2	2 8 5 1 2 6 8 2 9
Example-2	1 3 5 6 8	5	5 6 8 1 3
Example-3	9 2 5 7	6	9 2 5 7