Practice Exercise #31: Web Browser History

http://www.comp.nus.edu.sg/~cs1020/4_misc/practice.html

(This exercise is Sit-in Lab #3 Set A of AY2012/2013 Semester 2)

Objective: Using LinkedList or Stack class

Background

In a modern web browser, web pages visited are stored in a browsing history and the user can navigate to any page in the browsing history by using the "forward" and "back" buttons on the web browser. In this task, you are to implement a simple simulation of these operations.

Note: You are required to make use of the LinkedList ADT or any of its variants, or the Java API Stack, in solving this problem.

Task Statement

The behaviours of the "forward" and "back" buttons in a web browser are described as follows.

A user can browse a page by entering a web address, known as the URL (Universal Resource Locator). The URL entered is stored in a browsing history list. The user may also click the "forward" or "back" buttons to navigate to a different page in the browsing history. At any time, when the "back" button is clicked, the browser is redirected to the previous page that the user visited. The "back" button can be clicked many times to redirect the browser to a page further back in history. When the browsing history is empty or the current page is the earliest page in the browsing history, clicking the "back" button will do nothing.

The "forward" button exhibits the opposite behaviour to the "back" button. When clicked, the browser redirects to the next URL in the browsing history. Clicking the "forward" button will do nothing if the browsing history is empty or the current page is the latest page in the browsing history.

Note that each time the user navigates to a new page by entering a URL, the URLs in the browsing history that come after the current URL are cleared. An example is given below.

Browsing history (ordered from earliest to latest):

```
http://www.google.com
http://www.gmail.com ← current page user is browsing
http://www.comp.nus.edu.sg
http://www.yahoo.com
```

If the user then enters a new URL – www.nus.edu.sg, the browsing history is changed as follows:

```
http://www.google.com
http://www.gmail.com
http://www.nus.edu.sg ← current page
```

Input

The input consists of some lines, where each line consists of exactly one of the following 3 types of instructions.

- If the line consists of the word "BACKWARD", it indicates the "back" button has been clicked.
- If the line consists of the word "FORWARD", it indicates that the "forward" button has been clicked.
- Otherwise, the line consists of a URL that the user has entered.

Note that the number of lines is not indicated in the input. An example input is given below.

```
http://www.google.com
http://www.hotmail.com
BACKWARD
FORWARD
http://www.comp.nus.edu.sg
https://ivle.nus.edu.sg/
BACKWARD
BACKWARD
http://uva.onlinejudge.org/
BACKWARD
```

Output

The final browsing history at the end of the simulation ordered by earliest page to latest page, as well as the current page that is being browsed. The output for the example input above is given below:

```
Browsing History:
http://www.google.com
http://www.hotmail.com
http://uva.onlinejudge.org/
Current Page:
http://www.hotmail.com
```

If the browsing history is empty, simply print:

```
Browsing history is empty.
```

Skeleton Program

Your program is to be named **WebBrowser.java**. A skeleton program is provided, but you can change it any way or add new class(es) as you deem fit when you write your program. In the skeleton program, the ListNode <E> class is provided as well as a very rudimentary LinkedList <E> class (if you want to use this class you will have to complete it). If you choose to use the **Stack** API, you may remove the code related to **ListNode** and **LinkedList** from your program.