Lab 06 - Nodes & Stacks Problems

Direction: Submit typed work in the Labs directory of your github repositor or dropbox, or upload to the google classroom assignment. Each part should be a separate files. The files named should be `"lab6A.cpp"` and `"lab6B.cpp"` respectively.

Part A: In class

Your objective is to write the definition of the function `BackwardRotation()` whose header is

```
template<typename T>
void BackwardRotation(Node<T>*& root)
```

Given that *root* is referencing a null-terminated doubly linked list, the function makes the first node of the linked list referenced by *root* the new last node of the linked list referenced by *root* given that the linked list contains at least two nodes.

Part B: Take home

Your objective is to write the definition of the following functions

- ☐ the function `IsValidWord()` whose header is

  ```
  bool IsValidWord(string wrd)
  ```

  It returns true, if *wrd* consists only of As and Bs and it contains the substring `"aab"` where a and b can be in any case; otherwise, it returns false. For instance, the function calls `IsValidWord("aAabbB")` and `IsValidWord("aBAb")` will evaluate to true and false respectively. You must use a stack.

- ☐ the function `SameDifference()` whose header is

  ```
  int SameDifference(Node<int>* root,int k)
  ```

  It returns the number of pairs of distinct nodes of the linked list reference by *root* whose absolute value difference is equal to k. If k is negative or the linked list has less than two nodes, the functions returns 0.