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 `ForwardRotation()` whose header is

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

Given that *root* is referencing a null-terminated doubly linked list, the function makes the last node of the linked list referenced by *root* the new first 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* consist of the same amount of As and Bs with no other characters where a and b can be in any case; otherwise, it returns false. For instance, the function calls `IsValidWord("aAabbB")` and `IsValidWord("aBA")` will evaluate to true and false respectively. You must use a stack.

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

  ```
  int DivisibleSum(Node<int>* root,int m)
  ```

  It returns the number of pairs of distinct nodes of the linked list reference by *root* whose sum is divisible by m. If m is not positive or the linked list has less than two nodes, the functions returns 0.