Programming Assignment #4

Recursion practice

CS 3358.751, Summer II 2013
Instructor: Jill Seaman

Due: Monday, 7/29/2012 (upload electronic copy by 11:30am)

Problem:

Write functions for each of the following problems. Each problem should be solved by writing a recursive function (potentially with an auxiliary/driver function). Your final program should not have any loops in it.

All of your solutions should be in a single .cpp file. The main function of the file should demonstrate each of your solutions, by running some tests and producing some output.

1. **Table of Squares:**
   Convert the following function to one that uses recursion:

   ```
   void tableOfSquares (int n) {
      for (int num=1; num<=n; num++) {
         cout << num << "      " << (num * num) << endl;
      }
   }
   ```

2. **Recursive Power Function**
   Write a function that uses recursion to raise a number to a power. The function should take two arguments, the number to be raised to the power (floating point) and the power (a non-negative int).

3. **isMember function**
   Write a boolean function named isMember that takes three arguments: an array, its size (number of elements) and a value. It should return true if the value is found in the array, or false if the value is not found in the array.

4. **maxNode function**
   Write a function that takes one argument, a List_3358<Item> (your program will need to include the appropriate header file). The function should return the largest Item in the list. Your function should fail if the list is empty. (Hint: maybe try writing this for an array or linked list of ints first, then converting to List_3358).

5. **Palindrome detector**
   A palindrome is any word, phrase, or sentence that reads the same forwards or backwards. Here are some palindromes (find more with google):

   level
   Pot top
   A man a plan a canal Panama

   Write a boolean function that determines if a string argument is a palindrome. The function should return true if the argument reads the same forwards and backwards. For full credit, your function should ignore spaces and be case-insensitive. Assume the input is just letters and spaces. Hint: you can do recursive calls on **any** substring of the original one (as long as it is shorter, and you have enough base cases).

Output:

Here is the output from my file. I'm sure you can do better:

```
Table of squares:
N      N Squared
1        1
2        4
3        9
4        16
5        25
6        36
7        49
8        64
9        81
10       100

Power function:
2 to the 5th power: 32

Member function
1 2 3 4 5 has 4?: YES
1 2 3 4 5 has 6?: NO

MaxNode of {10,101,11,9,100): 101

Is "A man A plan A canal Panama" a palindrome? YES
Is "No one" a palindrome? NO
```

NOTES:

- Just one file. 5 (or more) functions plus main for testing.

Style:

See the Style Guidelines document on the course website.

Logistics:

Please submit you solution in a single file. You can call it recursion_xxxxxx.cpp.

The xxxxx is your TX State NetID (your txstate.edu email id).

You don’t need to submit the List_3358.h file. I’ll find one to use with it.

Submit: an electronic copy only, using the Assignments tool on the TRACS website for this class.