Tutorial: Data Structures I – Linked Lists (contd.) and Stacks

1. Download the *SimpleArrayStackofchars.zip* file and re-write methods where required to answer the following questions:
    a. Re-write the Main method so that there is a call to another method called *reverseUsingPop* that will take an array of chars and print the array elements in reverse order using the *pop* method.
    b. Write a method called *reverseUsingForloop* that will take an array of chars and print the array elements in reverse order using the *For* loop. Comment on the similarities and differences with the approach in Q2 above.
    c. Create a new Class called *SimpleListStackofchars* that uses a linked list to store the stack instead of an array. Use it to add the following strings to the stack and then to print them out in a reverse order:
    "a","b","c","...","x","y","z","backwards","alphabet","the","is","this".

2. Write a simple class that will create a String array to hold up to 1000 banner ids of the format "000abcdef" where "abcdef" are six digits and each one is from 0-9. Each banner id that is created is to be placed into the array using an index. The index is determined by the integer made up of the last three digits of the banner id (i.e. the digits "def"). The problem is that there are likely to be repeated indexes so to handle the collisions use *linear probing.*
3. Repeat question 2 but this time do not use a String array, instead use an array where each element of the array is a single linked list and use *separate chaining* to handle the collisions of repeated indexes.