

Solve all Questions

Q1 [2]. Give the Big-Oh notation for the following functions

$f(N) = N^2 \log N^2 + 2N \log 2N$	
$f(N) = (N \cdot (100N + 5 + N^2))^2$	
$f(N) = N^{1/2} + \log \log N$	
$f(N) = 1000 \log \log N + \log N$	

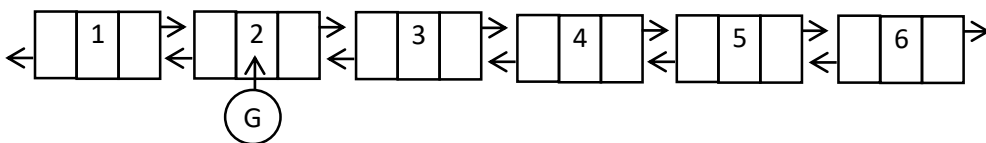
Q2 [4]. Given the following code, estimate the number of operations and describe the worst case running time in Big-Oh notation in terms of the variable n. Show trace where appropriate.

<pre>public int silly(int n, int m) { if (n < 1) return m; else return silly(n/2, m); }</pre>	<p>silly(8, 0)</p>
<pre>public void yoyo(int n) { for (int i = 0; i < 2 * n; i++) { if (i%2==0) System.out.println("even"); else System.out.println("odd"); } }</pre>	<p>yoyo(5)</p>

Q3. [4 points] Write a method **public Node getKthValue(Node H, int k)** that takes two parameters. Node pointer H points to the Head of a Singly Linked List; k is an integer value. Write java code for this method that return a Node pointer to the kth node. Start counting Head as position 1.

Q4. [2 points] Assume the following Data Structure was created using the Node class defined for the Doubly Linked List. Write code to swap the nodes containing values 2 and 5. Node G is currently pointing to a node containing value 2. You may create other pointers as necessary.

```
public class Node{
    Node prev, next;
    int value;
}
```



Q5. [3 points] Write a method `createArrayOfLL` that takes a Scanner to a text file containing only words (English), one in each line. This method creates an array of 26 linked lists, fills the arrays with all words from this file and return a reference to this array. Use the following points to write your method.

1. Create an array of 26 Singly Linked Lists.
2. Use scanner to read a word from the file. Insert this word into the array based on its first letter, eg: any word that begins with 'a' goes to A[0] list etc. Note, there are 26 characters in English language, hence 26 linked lists.
3. Return the reference to the array of linked lists

You must consider ALL cases . You must write reasonable comments to describe your algorithm. You may assume public fields for each node, data and next.

Note: You do not have to provide java code. A clearly explained algorithm is enough.