

ICS H22
Final Exam
Fall 2003

Instructor: Sandy Irani
Friday, December 12, 2003

Complete all of the following questions. There are a total of 100 points.

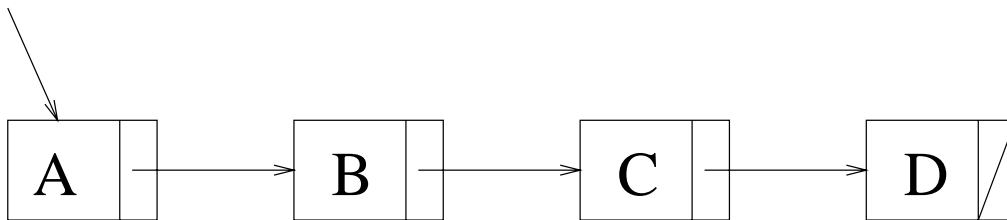
1	
2	
3	
4	
5	

6	
7	
8	
9	
Total	

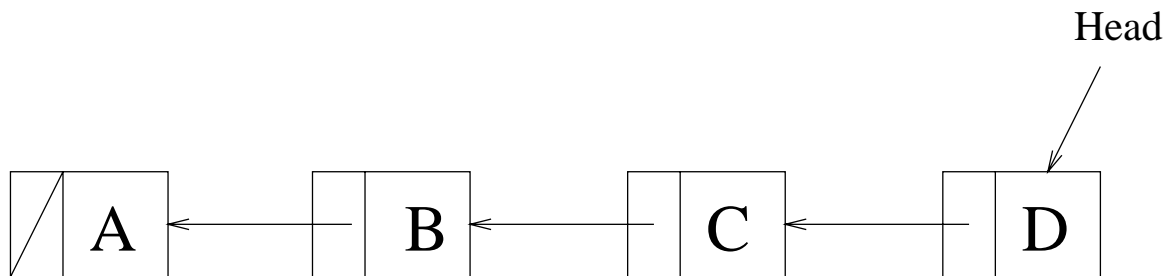
1. (15 points) Suppose you have a class `LinkedList` which maintains a singly linked list. Each element in the list is an object of class `ListNode` given below. The `LinkedList` class has a `head` reference which points to the first item in the list. Write a method which will reverse the list.

That is, if the linked list looks like this before the call to `reverseList()`:

Head



then it should look like this after the call to `reverseList()`:



Your method should be as efficient as possible in terms of the extra memory used and the time it takes to run. Specifically, you should use only $O(1)$ extra space. Give the running time (in O -notation) of your method.

```
public class ListNode
{
    Object element;
    ListNode next;
}
```

```
public class LinkedList
{
    ListNode head;

    public void reverseList()
    {
```

```
    }
}
```

2. (10 points) Below is a pseudo-code to calculate the n^{th} Fibonacci number f_n .

- (a) Give a recurrence relation describing $T(n)$ the running time for the method on input n . Note you do not have to solve the recurrence relation.

```
public static int CalculateFibonacci( int n )
{
    if ( n == 0 || n == 1 )
        return 1;

    else
        return( CalculateFibonacci( n-1 ) + CalculateFibonacci( n-2 ) );
}
```

- (b) Is this an effective use of recursion? Why or why not?

3. (15 points) Write a method which will take in an instance of class `PostfixExpression` and evaluate the expression. The expression consists of a list of tokens which are either operators or operands. Your method should output an appropriate error message if it encounters a problem with the input. You have the following classes at your disposal:

```
// an operand or an operator
public class Token {

    public boolean isOperator(); // returns true if the token is an operator

                                // the following four methods return false
                                // if the token is an operand
    public boolean isPlus();     // returns true if token is the + operator
    public boolean isMinus();    // returns true if token is the - operator
    public boolean isMult();     // returns true if token is the * operator
    public boolean isDiv();      // returns true if token is the / operator

    public float   getValue();   // returns the value of an operand if it
                                // is a token. Returns 0 otherwise.
}

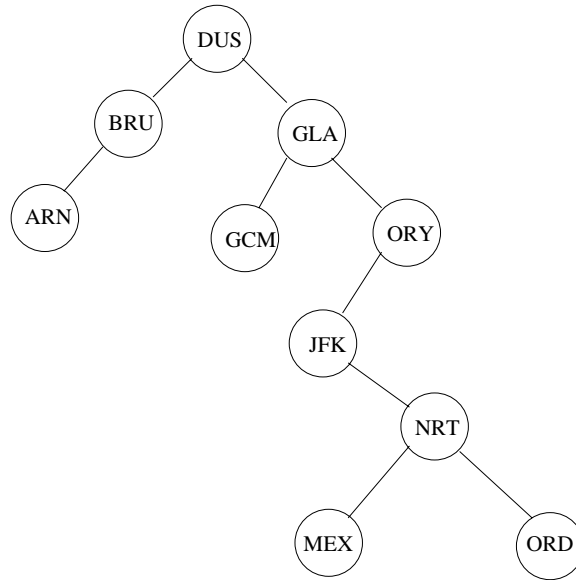
public class PostfixExpression {

    public void reset();         // resets current token to be first token
    public Token getNextToken(); // returns the current token in the expression
                                // and advances to the next token.
                                // if there are no remaining tokens, returns
                                // null
}

// a stack that can hold tokens
public class StackOfTokens {

    public StackOfTokens();     // creates an empty stack
    public Token pop();         // pops and returns token at top of stack
    public void push( Token t ); // pushes t on top of stack
}
```

4. (5 points) Consider the following binary search tree below.



Using the algorithm shown in class, show what the binary search tree would look like after the node labelled GLA is deleted.

6. (a) (5 points) What is the value returned by each of the following Scheme expressions?

```
(* (- 3 4) (+ 5 6))
```

```
(first (rest (rest '(Have A Happy New Year))))
```

```
(* 100 (if (odd? (first (rest '(17 77 12)))) 33 55))
```

(b) (5 points) Suppose you have the following definitions

```
(define filter
  (lambda (test? L)
    (cond
      ((null? L) '())
      ((test? (first L)) (filter test? (rest L)))
      (else (cons (first L) (filter test? (rest L)))))))
```

```
(define OOP?
  (lambda (LanguageDescription)
    (equal? 'OOP (first LanguageDescription))))
```

i. What is the value returned by the following expressions?

```
(filter even? '(31 34 55 34 12 3 101))
```

ii. What is the value returned by this expression?

```
(filter OOP? '((Functional Scheme) (OOP Eiffel) (Functional ML)
               (Imperative Psacal) (OOP Java)))
```

7. (10 points) A valid password is defined to be a string of characters whose length is four, five or six characters. Furthermore, the string must contain a character which is not a letter. Give a finite automaton to determine whether a string is a valid password. You can assume the string is read character at a time and it is determined whether each character is a letter or a non-letter. Thus, the input alphabet to your automaton is $\{L, N\}$ (L for letter and N for non-letter). Give your answer in graph form, where the nodes (circles) represent states and arrows represent transitions. Denote the accepting states with a double circle. (Note there can be more than one accepting state). Clearly indicate which state is the start state.

8. (10 points) Give a big-Oh notation, in terms of n , of the running time of each of the loops shown below.

(a) $s \leftarrow 0$
for (int $i = 0$; $i < n$; $i++$)
 $s \leftarrow s + i^2$

(b) $s \leftarrow 0$
for (int $i = 0$; $i < n$; $i++$)
 for (int $j = 0$; $j < n$; $j++$)
 $s++$

(c) $s \leftarrow 0$
for (int $i = 0$; $i < n$; $i++$)
 $s++$
for (int $j = 0$; $j < n$; $j++$)
 $s++$

(d) $s \leftarrow 0$
for (int $i = 1$; $i < n$; $i = i * 2$)
 $s++$

(e) $s \leftarrow 0$
for (int $i = 1$; $i * i < n$; $i++$)
 $s++$

9. (10 points) What is the output of the function `test()`?

```
class TestParameterPassing
{
    public void test()
    {
        int[] a = new int[ 10 ];
        int[] b = new int[ 10 ];

        for ( int i = 0; i < a.length; i++ )
        {
            a[ i ] = i;
            b[ i ] = -i;
        }

        System.out.println( a[ 2 ] );
        change1( a );
        System.out.println( a[ 2 ] );
        change2( a[ 2 ] );
        System.out.println( a[ 2 ] );
        change3( a );
        System.out.println( a[ 2 ] );
        a = b;
        System.out.println( a[ 2 ] );
    }

    public void change1( int[] b )
    {
        for ( int i = 0; i < b.length; i++ )
            b[ i ] = b[ i ] * b[ i ];
    }

    public void change2( int b )
    {
        b = b * b;
    }

    public void change3( int[] b )
    {
        int[] a = new int[ b.length ];

        for ( int i = 0; i < b.length; i++ )
            a[ i ] = b[ i ] * b[ i ];
        b = a;
    }
}
```