Introduction to Computers and Programming (CSL101) Major May 5, 2004

Name: **Entry:** **Grp:**

Note: Maximum marks : 120 (a mark a minute). All notations standard (as done in class). Answer only in the space provided. 4 pages in all.

Question #	Max. Marks	Marks Obtained
1	30	
2	30	
3	30	
4	30	
Total	120	

1. **[30 Marks]** Consider the representation of a polynomial

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + ... + a_{n-1}x^{n-1} + a_nx^n \qquad (1)$$

of degree n stored in an array $A[0..n]$ with $A[i]$ holding a_i (the coefficient of x^i). Write an algorithm that takes a parameter x and returns the value of the polynomial $f(x)$ under the following constraint: THE TOTAL NUMBER OF MULTIPLICATIONS SHOULD BE NO MORE THAN n. The a_i's and x are real numbers. Continue at the back of this page.

[Hint: The polynomial $1 + 5x + 3x^2$ can be evaluated using just 2 multiplications.]

2. **[30 marks]** Develop an abstract data-type called set in Java . Represent a set of objects as a list (there should be no duplications) and develop methods for:

   (a) add, for adding a new element to a set.

   (b) belongs, for checking whether an element belongs to a set.

   (c) intersection, for finding the intersection of two sets.

   (d) union, for finding the union of two sets.

   Make the implementations for *union* and *intersection* as efficient as possible.

   You may assume that the following List class is available with the public methods which you can use.

```
public class List {
   public List(); // creates an empty list
   public static List newList(List ls); // returns a new list
   public int head(); // returns the first element
   public List tail(); // returns the sublist without the first element
   public void attach(int x); // attach an object at the start
   public boolean isempty(); // checks whether the list is empty
   public int length(); //returns the length of the list
   public static List reverse(List ls);
   public static List append(List l1, List l2);
}
```

   Assume that you are buiding a set of integer elements. Continue at the back of this page.

3. **[30 Marks]** In a modified binary search algorithm where we search for element x in a sorted array $A[0..n-1]$, we employ the following strategy:

   - Divide the range into two parts *left* and *right*, where *left* contains the left 1/4 elements of A, and *right* holds the right 3/4 elements of A.
   - Discard the part whose range does not contain x.
   - Proceed with the search recursively within the part that was not discarded.

   Determine the worst case time complexity of this search procedure. Is the order better or worse than the binary search procedure where we divide the range into two equal halves?

4. **[30 Marks]** The definition for the Fibonacci numbers can be written in the matrix notation as follows:

$$\begin{bmatrix} f_n \\ f_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f_{n-1} \\ f_{n-2} \end{bmatrix} \text{ for } n > 1$$

where $f_0 = 0$ and $f_1 = 1$. Using the above relations and the 'fast power' algorithm discussed in the class develop an ML program for computing the n^{th} Fibonacci number in $O(\log n)$ steps. Assume that the function $createmat(a, b, c, d)$ returns the matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, and the function $access(mat, i, j)$ returns the $(i, j)^{th}$ element of the matrix mat. (that is, $access(mat, 0, 0)$ returns a, and so on). Also, note that $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.