Data Structures
Spring 2009, Midterm Exam, March 31
7:00-9:00 PM

1. (15%) A palindrome is a string that reads the same forward and backward. For instance, the following are palindromes: dad, level, did, madamimadam. Write both recursive and iterative C (or C-like) functions, $check(A, left, right)$, which returns true if A is a palindrome. Let array A hold a string of n characters.

2. (10%) Write a recursive algorithm $HanoiTower(frompeg, topeg, auxpeg, n)$ to move n disks from $frompeg$ to $topeg$ using $auxpeg$ as an auxiliary. In the algorithm, $frompeg$, $topeg$ and $auxpeg$ are character type and n is an integer. The algorithm will produce a list of instructions to move the disks as output.

3. (10%) Given a sorted vector $A[N]$ with distinct elements. Write an $O(logn)$ algorithm that finds the interval of a given value x which is different from any elements in A. More specifically, the function should return i with
   (a) $i = 0$ if $x < A[0]$.
   (b) $i = N$ if $A[N-1] < x$.
   (c) Otherwise the returned i should satisfies $A[i-1] < x < A[i]$.

4. (10%) Let arrays A and B hold m and n sorted numbers, respectively. Write a pseudo code $merge(A, B, C, m, n)$, which merges the numbers in A and B and produces a sorted list in C. The worst cast time complexity of the algorithm is $O(m+n)$.

5. (10%) Assume we have an array A with dimensions $5 \times 3 \times 6$, with each element occupying a byte. Let 1069 be the address of A[3][2][3].
   (a) What is the address of $A[4][1][5]$ if we use the row-major representation?
   (b) Repeat the above question if we use the column-major representation?

6. (25%) Give answers to the following short questions. You do not need to prove them or show the intermediate results.
   (a) (5%) Give three applications that use the stack as an auxiliary data structure.
   (b) (5%) A queue is being maintained circularly in an array Q[n] with variables $front$ and $rear$ set being pointed to the beginning and ending elements. Give the conditions to test if the queue is empty and full, respectively.
   (c) (5%) In the above question, why can we only use $n-1$ elements to represent the queue?
   (d) (5%) Place the following functions into asymptotically ascending order: n, $n^2 logn$, n^2, $logn$, $log(n!)$, $n!$, 1.5^n.
   (e) (5%) Translate the following expression to its postfix notation: $a+b-c*d+(e+f)*g$

7. (10%) Please use a stack to indicate how to evaluate the following single digit numbers in postfix expression 6 2 / 3 − 4 2 * + step by step. (You should draw the stack status after each step.)

8. (10%) Give the worst case time complexity of the following procedures as a function of n by using "big oh" notation.

(a)
```
xxx (int : n )
  {
   int i, j, sum;
     sum = 0 ;
     for ( i = n/ 2 ; i >= 1; i − − ) {
        j = 2 ;
        while ( j < n )
           j = j * 2;
        sum = sum + j;
     }
  }
```

(b)
```
int yyy (int: n)
  {
     if ( n <= 1 )
        then return (1);
        else return (yyy(n/3) + yyy(n/3) + yyy(n/3) );
  }
```