1.1 Solve the recurrence equations

$T(n) = T(n-1) + n$

$T(1) = 1$

1.2 What is the generating function G(z) for the sequence of Fibonacci numbers?

1.3 $\sum_{1 \le k \le n} O(n)$, where O(n) stands for order n is

(a) $O(n)$ (b) $O(n^2)$

(c) $O(m^3)$ (d) $O(3n^2)$

1.4 Consider the following two functions:

$$g_1(n) = \begin{cases} n^3 \text{ for } 0 \le n \le 10{,}000 \\ n^2 \text{ for } n \ge 10{,}000 \end{cases}$$

$$g_2(n) = \begin{cases} n \text{ for } 0 \le n \le 100 \\ n^3 \text{ for } n > 100 \end{cases}$$

Which of the following is true?

(a) $g_1(n)$ is $O(g_2(n))$ (b) $g_1(n)$ is $O(n^3)$

(c) $g_2(n)$ is $O(g_1(n))$ (d) $g_2(n)$ is $O(n)$

1.5 Which of the following is false?

(a) $100n \log n = O\left(\frac{n \log n}{100}\right)$

(b) $\sqrt{\log n} = O(\log \log n)$

(c) If $0 < x < y$ then $n^x = O(n^y)$

(d) $2n \ne O(nk)$

1.6 The concatenation of two lists is to be performed in O(1) time. Which of the following implementations of a list should be used?

(a) Singly linked list

(b) Doubly linked list

(c) Circular doubly linked list

(d) Array implementation of list

1.7 Let $f(n) = n^2 \log n$ and $g(n) = n(\log n)^{10}$ be two positive functions of n. Which of the following statements is correct?

(a) $f(n) = O(g(n))$ and $g(n) \ne O(f(n))$

(b) $g(n) = O(f(n))$ and $f(n) \ne O(g(n))$

(c) $f(n) \ne O(g(n))$ and $g(n) \ne O(f(n))$

(d) $f(n)=O(g(n))$ and $g(n) = O(f(n))$

1.8 In the worst case, the number of comparisons needed to search singly linked list of length n for a given element is

(a) $\log_2 n$ (b) $n/2$

(c) $\log_2 n - 1$ (d) n

1.9 Consider the following functions

$f(n) = 3n^{\sqrt{n}}$

$g(n) = 2^{\sqrt{n}\log_2 n}$

$h(n) = n!$

Which of the following is true?

(a) h(n) is O(f(n)) (b) h(n) is O(g(n))

(c) g(n) is not O(f(n)) (d) f(n) is O(g(n))

1.10 Consider the following algorithm for searching for a given number x in an unsorted array A[1..... n] having n distinct values:

1. Choose an i uniformly at random from 1...n;
2. If A[i] = x then Stop else Goto 1;

Assuming that x is present on A, what is the expected number of comparisons made by the algorithm before it terminates?

(a) n (b) n – 1

(c) 2n (d) n/2

1.11 The running time of the following algorithm

Procedure A(n)

If n < = 2 return (1) else return $\left(A\left(\lceil \sqrt{n} \rceil\right)\right)$;

is best described by

(a) O(n) (b) O(log n)

(c) O(log log n) (d) O(1)

1.12 Consider the following three claims

1.$(n + k)^m = \Theta(n^m)$, where k and m are constants

2.$2^{n+1} = O(2^n)$

3.$2^{2n+1} = O(2^n)$

Which of these claims are correct?

(a) 1 and 2
(b) 1 and 3
(c) 2 and 3
(d) 1, 2 and 3

1.13 Consider the following C function.

```
float f (float x, int y)
{
  float p, s; int i;
  for (s = 1, p = 1, i = 1; i < y; i ++)
  {
    p* = x/i;
    s + = p;
  }
  return s;
}
```

For large values of y, the return value of the function f best approximates

(a) x^y (b) e^x
(c) $ln(1 + x)$ (d) x^x

1.14 The cube root of a natural number n is defined as the largest natural number m such that $m^3 \leq n$. The complexity of computing the cube root of n (n is represented in binary notation) is

(a) O(n) but not $O(n^{0.5})$
(b) $O(n^{0.5})$ but not $O((\log n)^k)$ for any constant $k > 0$.
(c) $O((\log n)^k)$ for some constant $k > 0$, but not $O((\log \log n)^m)$ for any constant $m > 0$
(d) $O((\log \log n)^k)$ for some constant $k > 0.5$, but not $O((\log \log n)^{0.5})$

1.15 The tightest lower bound on the number of comparisons, in the worst case, for comparison-based sorting is of the order of

(a) n
(b) n^2
(c) nlogn
(d) $n\log^2 n$

1.16 What does the following algorithm approximate? (Assume $m > 1, \varepsilon > 0$).

```
x = m;
y = 1;
while (x – y > ε)
{
  x = (x+y)/2;
  y = m/x;
}
    print (x);
```

(a) log m (b) m^2
(c) $m^{1/2}$ (d) $m^{1/3}$

1.17 Let A [1,..., n] be an array storing a bit (1 or 0) at each location, and f(m) is a function whose time complexity is $\Theta(m)$. Consider the following program fragment written in a C like language:

```
counter = 0;
for(i = 1; i < n; i++)
{
    if (A[i] = = 1) counter ++;
    else
    {
      f (counter);
      counter = 0;
    }
}
```

The complexity of this program fragment is

(a) $\Omega(n^2)$ (b) $\Omega(n \log n)$ and $O(n^2)$
(c) $\Theta(n)$ (d) $O(n \log n)$

1.18 The time complexity of the following C function is (assume $n > 0$)

```
int recursive (int n)
{
  if (n = = 1)
    return (1);
  else
    return(recursive(n–1)+ recursive(n–1));
}
```

(a) O(n) (b) O (n log n)
(c) $O(n^2)$ (d) $O(2^n)$

1.19 The recurrence equation:

$T(1) = 1$

$T(n) = 2T(n-1) + n, n \geq 2$

evaluates to

(a) $2^{n+1} - n - 2$ (b) $2^n - n$
(c) $2^{n+1} - 2n - 2$ (d) $2^n + n$

1.20 Let f(n), g(n) and h(n) be functions defined for positive integers such that $f(n) = O(g(n), g(n)) \neq O(f(n)), g(n) = O(h(n))$, and $h(n) = O(g(n))$. Which one of the following statements is FALSE?

(a) $f(n) + g(n) = O(h(n)) + h(n))$
(b) $f(n) = O(h(n))$
(c) $h(n) \neq O(f(n))$
(d) $f(n)h(n) \neq O(g(n)h(n))$

1.21 The time complexity of computing the transitive closure of a binary relation on a set of n elements is known to be

(a) $O(n)$ (b) $O(n \log n)$
(c) $O(n^{3/2})$ (d) $O(n^3)$

1.22 Let T(n) be a function defined by the recurrence $T(n) = 2T(n/2) + \sqrt{n}$ for $n \geq 2$ and $T(1) = 1$. Which of the following statements is TRUE?

(a) $T(n) = \Theta(\log n)$ (b) $T(n) = \Theta(\sqrt{n})$
(c) $T(n) = \Theta(n)$ (d) $T(n) = \Theta(n \log n)$

1.23 Suppose $T(n) = 2T(n/2) + n, T(0) = T(1) = 1$
Which one of the following is FALSE?

(a) $T(n) = O(n^2)$ (b) $T(n) = \Theta(n \log n)$
(c) $T(n) = \Omega(n^2)$ (d) $T(n) = O(n \log n)$

Common Data for Q. 1.24 & Q. 1.25

Consider the following C function:

```
double foo(int n)
{
  int i;
  double sum;
  if (n = = 0) return 1.0;
  else
  {
    sum = 0.0;
    for (i = 0; i < n ; i ++)
    sum + = foo(i);
    return sum;
  }
}
```

1.24 The space complexity of the above function is

(a) $O(1)$ (b) $O(n)$
(c) $O(n!)$ (d) $O(n^n)$

1.25 The space complexity of the above function is foo( ) and store the values of foo(i), $0 < = i < n$, as and when they are computed. With this modification, the time complexity for function foo( ) is significantly reduced. The space complexity of the modified function would be

(a) $O(1)$ (b) $O(n)$
(c) $O(n^2)$ (d) $O(n!)$

1.26 Consider the following C-program fragment in which i, j, and n are integer variables.
for (i = n, j = 0; i > 0; i / = 2, j + = i);
Let Val (j) = denote the value stored in the variable j after termination of the for loop. Which one of the following is true?

(a) val (j) = Θ (log n) (b) val (j) = Θ ($\sqrt{n}$)
(c) val (j) = Θ (n) (d) val (j) = Θ (n log n)

1.27 Consider the following is true?

$$.T(n) = 2T\left(\left\lceil \sqrt{n} \right\rceil\right) + 1, T(1) = 1$$

Which one of the following is true?

(a) $T(n) = \Theta(\log \log n)$ (b) $T(n) = \Theta(\log n)$
(c) $T(n) = \Theta\left(\sqrt{n}\right)$ (d) $T(n) = \Theta(n)$

1.28 Consider the following segment of C code

```
int j, n;
j = 1;
while (j < = n)
j = j* 2;
```

The number of comparisons made in the execution of the loop for any $n > 0$ is

(a) $\lceil \log_2 n \rceil + 1$ (b) n
(c) $\lceil \log_2 n \rceil$ (d) $\lfloor \log_2 n \rfloor + 1$

1.29 What is the time complexity of the following recursive function :

```
int DoSomething ( int n)
{
  if (n < = 2)
    return 1;
  else
    return DoSomething (floor (sqrt (n)))+n;
}
```

(a) $\Theta(n^2)$ (b) $\Theta(n \log_2 n)$
(c) $\Theta(\log_2 n)$ (d) $\Theta(\log_2 \log_2 n)$

1.30 An array of n numbers is given, where n is an even number. The maximum as well as the minimum of these n numbers needs to be determined. Which of the following is true about the number of comparisons needed?

(a) At least $2n - c$ comparisons, for some constant c, are needed.
(b) At most $1.5n - 2$ comparisons are needed.
(c) At least $n \log_2 n$ comparisons are needed.
(d) None of the above

1.31 Consider the following C code segment :

```
int IsPrime(n)
{
  int i, n;
  for(i = 2; i < = sqrt(n); i ++)
  {
    if (n % i = = 0)
    { printf ("Not Prime\n");
    return 0; }
  }
  return 1;
}
```

Let T(n) denote the number of times the for loop is executed by the program on input n. Which of the following is TRUE?

(a) $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(\sqrt{n})$
(b) $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(1)$
(c) $T(n) = O(n)$ and $T(n) = \Omega(\sqrt{n})$
(d) None of these

1.32 Arrange the following functions in increasing asymptotic order:

A. $n^{1/3}$ **B.** e^n
C. $n^{7/4}$ **D.** $n \log^9 n$
E. 1.0000001^n

(a) A, D, C, E, B (b) D, A, C, E, B
(c) A, C, D, E, B (d) A, C, D, B, E

1.33 When $n = 2^{2k}$ for some $k \geq 0$, the recurrence relation $T(n) = \sqrt{2}\, T(n/2) + \sqrt{n}$, $T(1) = 1$ evaluates to :

(a) $\sqrt{n}\,(\log n+1)$ (b) $\sqrt{n}\log n$
(c) $\sqrt{n}\log\sqrt{n}$ (d) $n\log\sqrt{n}$

1.34 Consider the following functions:

$f(n) = 2^n$
$g(n) = n!$
$h(n) = n^{\log n}$

which of the following statements about the asymptotic behaviour of f (n), g(n), and h (n) is true?

(a) $f(n) = O(g(n))$; $g(n) = O(h(n))$
(b) $f(n) = \Omega(g(n))$; $g(n) = O(h(n))$
(c) $g(n) = O(f(n))$; $h(n) = O(f(n))$
(d) $h(n) = O(f(n))$; $g(n) = \Omega(f(n))$

1.35 The minimum number of comparison required to determine if an integer appears more than n/2 times in a sorted array of n integers is

(a) $\Theta(n)$ (b) $\Theta(\log n)$
(c) $\Theta(\log^2 n)$ (d) $\Theta(1)$

Common Data Questions Q.1.36 and Q.1.37

Consider the following C functions:

```
int f1(int n)
{
  if(n = = 0 || n = = 1) return n;
  else
    return (2* f1( n -1) + 3 * f1(n - 2));
}
int f2( int n)
{
  int i ;
  int X[N] , Y[N] , Z[N] ;
  X[1] = 1; Y[1] = 2; Z[1] = 3;
  for( i = 2; i < = n ; i ++ )
  {
    X[ i ] = Y[ i - 1] + Z[ i - 2] ;
    Y[ i ] = 2 * X[ i ] ;
    Z[ i ] = 3 * X[ i ];
  }
return X[n] ;
}
```

1.36 The running time of f1 (n) and f2(n) are

(a) $\Theta(n)$ and $\Theta(n)$ (b) $\Theta(2^n)$ and $O(n)$
(c) $\Theta(n)$ and $\Theta(2^n)$ (d) $\Theta(2^n)$ and $\Theta(2^n)$

1.37 f1(8) and f2(8) return the values

(a) 1661 and 1640 (b) 59 and 59

(c) 1640 and 1640 (d) 1640 and 1661

1.38 What is the number of swaps required to sort n elements using selection sort, in the worst case?

(a) $\Theta(n)$ (b) $\Theta(n \log n)$

(c) $\Theta(n^2)$ (d) $\Theta(n^2 \log n)$

1.39 The running time of an algorithm is represented by the following recurrence relation:

$$T(n) = \begin{cases} n & n \le 3 \\ T\left(\frac{n}{3}\right) + cn & \text{otherwise} \end{cases}$$

Which one of the following represents the time complexity of the algorithm?

(a) $\Theta(n)$ (b) $\Theta(n \log n)$

(c) $\Theta(n^2)$ (d) $\Theta(n^2 \log n)$

1.40 Two alternative packages A and B are available for processing a database having 10^k records. Package A requires $0.0001\ n^2$ time units and package B requires $10 n \log_{10} n$ time units to process n records. What is the smallest value of k for which package B will be preferred over A?

(a) 12 (b) 10

(c) 6 (d) 5

1.41 Let W(n) and A(n) denote respectively, the worst case and average case running time of an algorithm executed on an input of size n. Which of the following is ALWAYS TRUE?

(a) $A(n) = \Omega(W(n))$ (b) $A(n) = \Theta(W(n))$

(c) $A(n) = O(W(n))$ (d) $A(n) = o(W(n))$

1.42 The recurrence relation capturing the optimal execution time of the Towers of Hanoi problem with n discs is

(a) $T(n) = 2T(n-2) + 2$

(b) $T(n) = 2T(n-1) + n$

(c) $T(n) = 2T(n/2) + 1$

(d) $T(n) = 2T(n-1) + 1$

1.43 Consider the following function:

```
int unknown (int n)
{
  int i, j, k = 0;
  for (i = n/2; i <= n; i++)
    for (j = 2; j <= n; j = j*2)
      k = k + n/2;
  return (k);
}
```

The return value of the function is

(a) $\Theta(n^2)$ (b) $\Theta(n^2 \log n)$

(c) $\Theta(n^3)$ (d) $\Theta(n^3 \log n)$

1.44 Which one of the following correctly determines the solution of the recurrence relation with $T(1) = 1$?

$$T(n) = 2T\left(\frac{n}{2}\right) + \log n$$

(a) $\Theta(n)$ (b) $\Theta(n \log n)$

(c) $\Theta(n^2)$ (d) $\Theta(\log n)$

1.45 Suppose we have a balanced binary search tree T holding n-numbers. We are given two numbers L and H and wish to sum up all the numbers in T that lie between L and H. Suppose there are m such numbers in T.
If the tightest upper bound on the time to compute the sum is $O(n^a \log^b n + m^c \log^d n)$, the value of $a + 10b + 100c + 1000d$ is ____.

1.46 Consider the following C function.

```
int fun1 (int n)
{   int i, j, k, p, q = 0;
    for (i = 1; i<n; ++i)
    {   p = 0;
        for (j=n; j>1; j=j/2)
        ++p;
        for (k=1; k<p; k=k*2)
        ++q;
    } return q;
}
```

Which one of the following most closely approximates the return value of the function fun1?

(a) n^3 (b) $n(\log n)^2$

(c) $n \log n$ (d) $n \log(\log n)$

1.47 An algorithm performs $(\log N)^{1/2}$ find operations, N insert operations, $(\log N)^{1/2}$ delete operations, and $(\log N)^{1/2}$ decrease-key operations on a set of data items with keys drawn from a linearly ordered set. For a delete operation, a pointer is provided to the record that must be deleted. For the decrease-key operation, a pointer is provided to the record that has its key decreased. Which one of the following data structures is the most suited for the algorithm to use, if the goal is to achieve the best total asymptotic complexity considering all the operations?

(a) Unsorted array
(b) Min-heap
(c) Sorted array
(d) Sorted doubly linked list

1.48 Consider a complete binary tree where the left and the right subtrees of the root are max-heaps. The lower bound for the number of operations to convert the tree to a heap is

(a) $\Omega(\log n)$ (b) $\Omega(n)$
(c) $\Omega(n \log n)$ (d) $\Omega(n^2)$

1.49 An unordered list contains n distinct elements. The number of comparisons to find an element in this list that is neither maximum nor minimum is

(a) $\Theta(n \log n)$ (b) $\Theta(n)$
(c) $\Theta(\log n)$ (d) $\Theta(1)$

1.50 Consider the equality $\sum_{i=0}^{n} i^3 = X$ and the following choices for X

I. $\Theta(n^4)$ II. $\Theta(n^5)$
III. $O(n^5)$ IV. $\Omega(n^3)$

The equality above remains correct if X is replace by

(a) Only I
(b) Only II
(c) I or III or IV but not II
(d) II or III or IV but not I

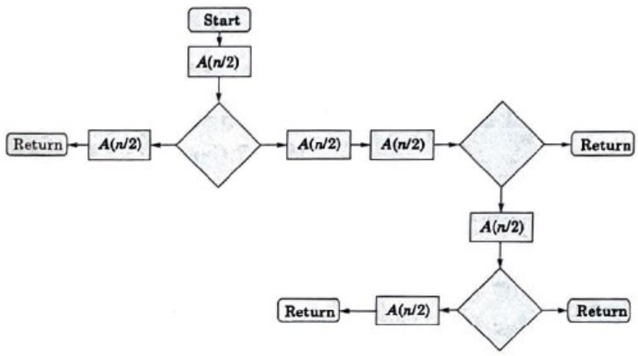
1.51 Let $f(n) = n$ and $g(n) = n^{(1+\sin n)}$, where n is a positive integer. Which of the following statements is/are correct?

I. $f(n) = O(g(n))$
II. $f(n) = \Omega(g(n))$

(a) Only I (b) Only II
(c) Both I and II (d) Neither I nor II

1.52 The given diagram shows the flow chart for a recursive function $A(n)$. Assume that all statements, except for the recursive calls, have $O(1)$ time complexity. If the worst case time complexity of this function is $O(n^{\alpha})$, then the least possible value (accurate up to two decimal positions) of α is ________.

Flow chart for Recursive Function $A(n)$



1.53 N items are stored in a sorted doubly linked list. For a *delete* operation, apointer is provided to the record to be deleted. For a *decrease-key* operation, a pointer is provided to the record on which the operation is to be performed. An algorithm performs the following operations on the list in this order: $\Theta(N)$ *delete*, $O(\log N)$ insert, $O(\log N)$ *find*, and $\Theta(N)$ *decrease-key*. What is the time complexity of all these operations put together?

(a) $O(\log^2 N)$ (b) $O(N)$
(c) $O(N^2)$ (d) $\Theta(N^2 \log N)$

1.54 Match the algorithms with their time complexities:

List-I (Algorithm)

(P) Towers of Hanoi with n disks
(Q) Binary search given n sorted numbers
(R) Heap sort given n numbers at the worst case
(S) Addition of two $n \times n$ matrices

List-II (Time complexity)

(i) $\Theta(n^2)$

(ii) $\Theta(n \log n)$

(iii) $\Theta(2^n)$

(iv) $\Theta(\log n)$

(a) P – (iii), Q – (iv), R – (i), S – (ii)

(b) P – (iv), Q – (iii), R – (i), S – (ii)

(c) P – (iii), Q – (iv), R – (ii), S – (i)

(d) P – (iv), Q – (iii), R – (ii), S – (i)

1.55 Consider the recurrence function

$$T(n) = \begin{cases} 2T(\sqrt{n}) + 1, & n > 2 \\ 2, & 0 < n \le 2 \end{cases}$$

Then $T(n)$ in terms of Θ notation is

(a) $\Theta(\log \log n)$ (b) $\Theta(\log n)$

(c) $\Theta(\sqrt{n})$ (d) $\Theta(n)$

1.56 Consider the following C function.

```
int fun (int n)
{   int i, j;
    for (i = 1; i <= n; i++)
    {  for (j = 1; j < n; j + = i)
       {
          printf{"%d %d", i, j);
       }
    }
}
```

Time complexity of fun in terms of Θ notation is

(a) $\Theta(n\sqrt{n})$ (b) $\Theta(n^2)$

(c) $\Theta(n \log n)$ (d) $\Theta(n^2 \log n)$

1.57 Consider the following functions from positive integers to real numbers:

$$10, \sqrt{n}, n, \log_2 n, \frac{100}{n}$$

Tire CORRECT arrangement of the above functions in increasing order of asymptotic complexity is:

(a) $\log_2 n, \frac{100}{n}, 10, \sqrt{n}, n$

(b) $\frac{100}{n}, 10, \log_2 n, \sqrt{n}, n$

(c) $10, \frac{100}{n}, \sqrt{n}, \log_2 n, n$

(d) $\frac{100}{n}, \log_2 n, 10, \sqrt{n}, n$

1.58 Let A be an array of 31 numbers consisting of a sequence of 0's followed by a sequence of 1's. The problem is to find the smallest index i such that $A[i]$ is 1 by probing the minimum number of locations in A. The worst case number of probes performed by an optimal algorithm is ________.

1.59 What is the worst case time complexity of inserting n elements into an empty linked list, if the linked list needs to be maintained in sorted order?

(a) $\Theta(n \log n)$ (b) $\Theta(n)$

(c) $\Theta(1)$ (d) $\Theta(n^2)$

1.60 What is the worst case time complexity of inserting n^2 elements into an AVL-tree with n elements initially?

(a) $\theta(n^4)$ (b) $\theta(n^2 \log n)$

(c) $\theta(n^3)$ (d) $\theta(n^2)$

1.61 For parameters a and b, both of which are $\omega(1)$, $T(n) = T(n^{1/a}) + 1$, and $T(b) = 1$. Then $T(n)$ is

(a) $\Theta(\log_2 \log_2 n)$ (b) $\Theta(\log_a \log_b n)$

(c) $\Theta(\log_b \log_a n)$ (d) $\Theta(\log_{ab} n)$

■■■■

Answers Algorithm Analysis and Asymptotic Notations

1.3 (b)	1.4 (a)	1.5 (b,d)	1.6 (c)	1.7 (b)	1.8 (d)	1.9 (d)	1.10 (a)	1.11 (c)
1.12 (a)	1.13 (b)	1.14 (c)	1.15 (c)	1.16 (c)	1.17 (c)	1.18 (d)	1.19 (a)	1.20 (d)
1.21 (d)	1.22 (c)	1.23 (c)	1.24 (b)	1.25 (b)	1.26 (c)	1.27 (b)	1.28 (d)	1.29 (d)
1.30 (b)	1.31 (b)	1.32 (a)	1.33 (a)	1.34 (d)	1.35 (b)	1.36 (b)	1.37 (c)	1.38 (a)
1.39 (a)	1.40 (c)	1.41 (c)	1.42 (d)	1.43 (b)	1.44 (a)	1.46 (d)	1.47 (a)	1.48 (a)
1.49 (d)	1.50 (c)	1.51 (d)	1.53 (c)	1.54 (c)	1.55 (b)	1.56 (c)	1.57 (b)	1.59 (d)
1.60 (b)	1.61 (b)							