((MARKS)) (1/2/3...)	1
((QUESTION))	Two main measures for the efficiency of an algorithm are
((OPTION_A))	Processor and memory
((OPTION_B))	Complexity and capacity
((OPTION_C))	Time and space
((OPTION_D))	Data and space
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following case does not exist in complexity theory
((OPTION_A))	Best case
((OPTION_B))	Worst case
((OPTION_C))	Average case
((OPTION_D))	Null case
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The Worst case occur in linear search algorithm when
((OPTION_A))	Item is somewhere in the middle of the array
((OPTION_B))	Item is not in the array at all
((OPTION_C))	Item is the last element in the array
((OPTION_D))	Item is the last element in the array or is not there at all

((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	.

((MARKS)) (1/2/3...)	1
((QUESTION))	The complexity of merge sort algorithm is
((OPTION_A))	O(n)
((OPTION_B))	O(log n)
((OPTION_C))	$O(n^2)$
((OPTION_D))	O(n log n)
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The input to a merge sort is 6,5,4,3,2,1 and the same input is applied to quick sort then which is the best algorithm in this case
((OPTION_A))	Merge sort
((OPTION_B))	Quick sort
((OPTION_C))	Both have same time complexity in this case as they have same running time
((OPTION_D))	Cannot be decided
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1

((QUESTION))	If there exists two functions f(n) and g(n). The constant c>0 and there exists an integer constant n_0>=1. If f(n)<=c*g(n) for every integer n>= n_0 then we say that____
((OPTION_A))	f(n)=O(g(n))
((OPTION_B))	f(n)=Θ (g(n))
((OPTION_C))	f(n)=Ω Ω (g(n))
f(n)=Θ (g(n))	f(n)=o(g(n))
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	Basic definition of big oh notation

((MARKS)) (1/2/3...)	1
((QUESTION))	In practice ______ is used to define tight upper bound on growth of function f(n)
((OPTION_A))	Big oh
((OPTION_B))	Big omega
((OPTION_C))	Big theta
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	The definition of big oh notation is f(n)<=c*g(n) which defines the upper bound on growth of the function f(n)

((MARKS)) (1/2/3...)	1
((QUESTION))	Examples of O(1) are ______
((OPTION_A))	Multiplying two numbers
((OPTION_B))	Assigning some value to a variable
((OPTION_C))	Displaying some integer on console
((OPTION_D))	All of the above

((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	All these operations are computed by single line expression evaluation

((MARKS)) (1/2/3...)	1
((QUESTION))	Examples of $O(n^2)$ algorithms are
((OPTION_A))	Adding two matrices
((OPTION_B))	Finding transpose of a matrix
((OPTION_C))	Initializing all elements of the matrix by 0
((OPTION_D))	All of the above
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	Within two for loops(nested), all these operations are performed.

((MARKS)) (1/2/3...)	1

((QUESTION))	Choose the correct time complexity of following code__
	
	while(n>0)
	{
	n=n/10
	}

((OPTION_A))	O(1)
((OPTION_B))	O(n)
((OPTION_C))	O(log n)
((OPTION_D))	$O(n^2)$
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The time complexity of binary search is______
((OPTION_A))	O(n)
((OPTION_B))	O(log n)
((OPTION_C))	O(n log n)
((OPTION_D))	O(n^2)
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	The list is divided at the mid and then the element is searched in either left half or right half.

((MARKS)) (1/2/3...)	1
((QUESTION))	Consider recurrence relation as T(0)=c1 T(n)=T(n-1)+c2 This can be expressed as
((OPTION_A))	O(n)
((OPTION_B))	O(log n)
((OPTION_C))	O(n log n)
((OPTION_D))	O(n^2)
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	T(n)=T(n-1)+c2 =T(n-2)+2c2 =T(n-3)+3c2 =T(n-k)+kc2 If k=n then T(n)=c1+nc2 Hence, T(n)=O(n)

((MARKS)) (1/2/3...)	1
((QUESTION))	Consider recurrence relation as T(0)=c1 and T(1)=c2 T(n)=T(n/2)+c3 This can be expressed as
((OPTION_A))	O(n)
((OPTION_B))	O(log n)
((OPTION_C))	O(n log n)
((OPTION_D))	O(n^2)
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Following is the method of solving recurrence relation
((OPTION_A))	Greedy method
((OPTION_B))	Backtracking
((OPTION_C))	Forward substitution method
((OPTION_D))	Divide and Conquer method
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The recurrence relation for factorial function is of the form ________
((OPTION_A))	T(n)=T(n-1)+c
((OPTION_B))	T(n)=T(n-1)+T(n-2)+c

((OPTION_C))	T(n/2)+c
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	A

((EXPLANATION)) (OPTIONAL)	The factorial function is as follows-
	fact(n)
	{
	if n=1
	return 1
	else
	return n * fact(n-1)
	}

((MARKS)) (1/2/3...)	1
((QUESTION))	The recurrence relation for fibonacci function is of the form ________
((OPTION_A))	T(n)=T(n-1)+c
((OPTION_B))	T(n)=T(n-1)+T(n-2)+c
((OPTION_C))	T(n/2)+c
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	B

((EXPLANATION)) (OPTIONAL)	The fibonacci function is as follows-
	fibb(n)
	{
	if n = = 0
	return 0
	if n = = 1
	return 1
	else
	return (fibb(n-1) + fibb(n-2))
	}

((MARKS)) (1/2/3...)	1

((QUESTION))	The frequency count of following code is____
	for(i=0;i<m;i++)
	{

	for(j=0;i<n;j++)
	{
	C[i][j]=a[i][j]+b[i][j];
	}
	}

((OPTION_A))	m + mn + mn
((OPTION_B))	m + n + mn
((OPTION_C))	m + n^2 + mn
((OPTION_D))	(m+1) + m(n+1) + mn
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Consider $T(n)=15n^3 + n^2 + 4$. Select the correct statement
((OPTION_A))	$T(n)=O(n^4)$
((OPTION_B))	$T(n)=\Omega\ \Omega\ (n^3)$
((OPTION_C))	$T(n)=\Omega\ \Omega\ (n^2)$
((OPTION_D))	All of the above
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1

((QUESTION))	Give the frequency count of 3rd Statement
	
	for(i=1;i<=n;i++)
	for(j=1;j<=i;j++)
	x=x+1;

((OPTION_A))	½(n^2+n)
((OPTION_B))	½(n^2+3n)
((OPTION_C))	n^2
((OPTION_D))	$(n+1)^2$
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	There are four algorithms for solving a problem. Their time complexities are O(n), O(n2), O(log n) and O(n log n). Which is the best algorithm?
((OPTION_A))	O(n)
((OPTION_B))	$O(n^2)$
((OPTION_C))	O(log n)
((OPTION_D))	O(n log n)
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The order of the recurrence relation $a_r-7a_{r-1}+10a_{r-2}=0$ is _______.
((OPTION_A))	3
((OPTION_B))	2
((OPTION_C))	1
((OPTION_D))	B
((CORRECT_CHOICE)) (A/B/C/D)	D

((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Characteristic roots of the recurrence relation a_r-2a_{r-1}+a_{r-2}=0 are ________
((OPTION_A))	1, -1
((OPTION_B))	-1, -1
((OPTION_C))	1, 1
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Charactristic polynomial of the recurrence relation b_n=-3b_{n-1}-b_{n-2} is _________.
((OPTION_A))	Z^2-3Z-2=0
((OPTION_B))	Z^2+3Z-2=0
((OPTION_C))	Z^2+3Z+2=0
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The general solution of the recurrence relation a_r-2a_{r-1}=0 is ______.
((OPTION_A))	a^r=c1$(-2)^r$

((OPTION_B))	$a^r=c2(2)^r$
((OPTION_C))	$a^r=c1(1)^r$
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Consider the recurrence relation, $a_n=a_{n-1}+2a_{n-2}$ with $a_9=3$ and $a_{10}=5$. Find a_7.
((OPTION_A))	1
((OPTION_B))	3
((OPTION_C))	5
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Charactristic polynomial of the recurrence relation $a_{r+2}-a_{r-2}=0$ is _______.
((OPTION_A))	Z-1=0
((OPTION_B))	$Z^2-1=0$
((OPTION_C))	$(Z-1)^2=0$
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	D

((EXPLANATION)) (OPTIONAL)	Given homogeneous recurrence relation can be written as $a_{r+2}+0a_{r+1}+0a_r+0a_{r-1}-a_{r-2}=0$ Order of this recurrence relation is 4. Hence characteristic polynomial is $Z^4-1=0$

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is not a homogeneous linear recurrence relation
((OPTION_A))	$a_{r+2}-a_{r-2}=0$
((OPTION_B))	$a_r=a_{r-1}+a_{r-2}$
((OPTION_C))	$a_r-2a_{r-1}=-a_{r-2}$
((OPTION_D))	$a_{r+3}+6a_{r+2}.a_{r+1}-4a_r=0$
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	If 4 and -1 are the characteristic roots of the recurrence relation then its homogeneous solution becomes ________
((OPTION_A))	$a^r=c1(-1)^r+c2(4)^r$
((OPTION_B))	$a^r=c0(-1)^r+c2$
((OPTION_C))	$a^r=(c1+c2.r)(-1)^r$
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	A recurrence relation of degree 1 is called _______.
((OPTION_A))	Linear
((OPTION_B))	Homogeneous
((OPTION_C))	Quadratic
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The generating function for the sequence 1, a, a^2, a^3, ….. is _______
((OPTION_A))	1/(1-z)
((OPTION_B))	1/(1-az)
((OPTION_C))	1/(1+az)
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	