1)Which of the following type of operators have higher precedence_____
a)Relational operators b)equality operators
c)Logical operators d)Arithmatic operators
ans-Arithmatic operators

2)Which of the following operators takes only integer operands?
a)+ b)/ c)% d)*
ans-%

3)Which of the operators associate from left?
a)+ b)> c)% d)all of these
ans-all of these

4)If a,b,c are integer variables with values 1,2,3 respectively, then what is the value of the expression
!((a+5)<(b+c))
a)0 b)6 c)5 d)1
ans-1

5) Give the value of x after execution of given code.
x=5
x=x++ +++x-x;
a)5 b)7 c)6 d)0
ans-7

6)Construct a logical expression to check whether x is largest among three numbers x,y,z
a)x>y&&x>z b)!(x<=y||x<=z) c)x>y,z d)both a and b
ans-d

Q.7 In a circular linked list…….

(A) components are all linked together in some sequential manner.

(B) there is no beginning and no end.

(C) components are arranged hierarchically.

(D) forward and backward traversal within the list is permitted.

Ans:B

Q.8 In a linked list with n nodes, the time taken to insert an element after an element pointed by some pointer is……..

(A) 0 (1) (B) 0 (log n) (C) 0 (n) (D) 0 (n 1og n)

Ans:A

Q.9 The data structure required to evaluate a postfix expression is …….

(A) queue (B) stack (C) array (D) linked-list

Ans:B

Q.10 Which of the following sorting methods would be most suitable for sorting a list which is almost sorted ……

(A) Bubble Sort (B) Insertion Sort

(C) Selection Sort (D) Quick Sort

Ans:A

Q.11 Representation of data structure in memory is known as…….

(A) recursive (B) abstract data type

(C) storage structure (D) file structure

Ans:B

Q.12 The largest element of an array index is called its……..

(A) lower bound. (B) range.

(C) upper bound. (D) All of these.

Ans. (C)

Q.13 Which data structure is used for implementing recursion?

(A) Queue. (B) Stack.

(C) Arrays. (D) List.

Ans. (B)

Q.14 Null character needs a space of……

A. zero bytes B. one byte

C. three bytes D. four bytes

Ans. (B)

Q.15 ……gives the step-by-step procedure for solving the problem which gives correct solution.

A.Algorithm B.Array

C.Link List D.None of the above

Ans.(A)

Q.16 Which of the followings are derived data types?

A.Array B.String

C.Float D.Both ‘a’ &’b’

Ans.(D)

Q.17 Which of the followings are application of data structure?

A.Facebook B.Searching

C.Sorting D.All of the above

Ans.(D)

Q.18 Which of the following data structure is not linear data structure?

A.Array B.Linked List

C.Both of above D.None of above

Ans.(A)

Q.19 Which of the following steps are correct for solving the problem?

A.Identify the problem B.Explore information & create ideas

C.Select the best ideas D.All of these

Ans.(D)

Q20) In how many blocks **Problem Analysis Chart**is divided?

A) 2 B)3 **C)4** D)5

Ans.(C)

Q21) **Interactivity Chart** is divided into subtasks called _____.

A) Function
B) Module
C) Sub program
D) None

Ans. (B)

Q 22) what is full form of IPO?

A) Input-Program-Output

B) Input-Parallel-Output

C)Input-Processing-Output

D) Inbuilt- Processing-Output

Ans.(C)

Q23) Algorithms are similar to____

A) Flowchart
B) Pseudo code
C) Interactivity chart
D) Problem Analysis Chart

Ans. (B)

Q24) Flowcharts can show errors in ____ which is not readily visible in the other charts.

A) Arithmetic operations
B) Logic
C) Code
D) All of these

Ans. (B)

Q25) Which equation is to be satisfied to find the BIG-O?

A) F(n)=c*g(n)
B) F(n)>=c*g(n)
C) F(n)<=c*g(n)
D) None

ANS:C

Q26) Time Complexity is

A) Time required for the machine to compile the program.
B) Time required for the machine to execute the program.
C) Time required for the machine to debug the program.
D) None

ANS:B

Q27) Step count for the following loop is

```
For(int i=5; i>0; i++)
```

A) 5 times
B) N+1 times
C) Infinite
D) No execution

ANS:C

Q28) which of the following is not an asymptotic notation?

A) BIG-O
B) Omega
C) Phi
D) Theta

ANS:C

Q29) What is the step count of 5^{th} line?

```
1. For(int i=0, i<4, i++)
2. {
3.   For(int j=5; j>0; j--)
4.   {
5.     Cout<<A[i][j];
6.   }
7. }
```

A) 16
B) 25
C) 20
D) 5

ANS:C

Q.30 Which of the following character used for null….?

a) /0 **b)\0** c)/n d)\n

Q.31 which of the following is not datatype…?

a)float b)long int c)long float **d)long string**

Q.32 Variable is……..

a) used to store a data value

b) used to store address

c) it has fixed value

d) none of these

Q.33 In c++ float datatype allocate……..memory.

a) 4byte **b) 8byte** c)16byte d)1byte

Q.34 In c float datatype allocate……..memory.

a) 8byte **b) 4byte** c)16byte d)1byte

Q.35 For function which is sequence of syntax…?

a) define-calling-declaration

b) declaration-define-calling

c) calling-define-declaration

d) none of these

Q.36. The function written by the user according to requirement known As…….

a) mathematical function b) string function

c) user define function d) none of above

Q.37. Which of the following is syntax for declaration of array…?

a) datatype array_name[size]

b) datatype string name[size]

c) datatype [size] array_name

d) none of these

Q38. Which of the following is type of function…?

a) mathematical function b) statistical function

c) utility function **d) all of above**

Q39. Array is…….datatype

a) homogenous datatype b) heterogenous datatype

c) both a & b d) none of above

Q40. The smallest element of an array index called……

a) lower bound b) upper bound c) range d) none of above

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

((QUESTION))	What will be the output of the program ?
	#include<stdio.h>
	
	int main()
	{
	int a[5] = {5, 1, 15, 20, 25};
	int i, j, m;
	i = ++a[1];
	j = a[1]++;
	m = a[i++];
	printf("%d, %d, %d", i, j, m);
	return 0;
	}

((OPTION_A))	2,1,15
((OPTION_B))	1,2,5
((OPTION_C))	3,2,15
((OPTION_D))	2,3,20
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	**Step 1**: int a[5] = {5, 1, 15, 20, 25}; The variable arr is declared as an integer array with a size of 5 and it is initialized to a[0] = 5, a[1] = 1, a[2] = 15, a[3] = 20, a[4] = 25 . **Step 2**: int i, j, m; The variable i,j,m are declared as an integer type. **Step 3**: i = ++a[1]; becomes i = ++1; Hence i = 2 and a[1] = 2 **Step 4**: j = a[1]++; becomes j = 2++; Hence j = 2 and a[1] = 3. **Step 5**: m = a[i++]; becomes m = a[2]; Hence m = 15 and i is incremented by 1(i++ means 2++ so i=3) **Step 6**: printf("%d, %d, %d", i, j, m); It prints the value of the variables i, j, m Hence the output of the program is 3, 2, 15

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

((QUESTION))	In C, if you pass an array as an argument to a function, what actually gets passed?
((OPTION_A))	Value of elements in array
((OPTION_B))	First element of the array
((OPTION_C))	Base address of the array
((OPTION_D))	Address of the last element of array
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	The statement 'C' is correct. When we pass an array as a funtion argument, the base address of the array will be passed.

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following statements are correct about 6 used in the program? int num[6]; num[6]=21;
((OPTION_A))	In the first statement 6 specifies a particular element, whereas in the second statement it specifies a type.
((OPTION_B))	In the first statement 6 specifies a array size, whereas in the second statement it specifies a particular element of array.
((OPTION_C))	In the first statement 6 specifies a particular element, whereas in the second statement it specifies a array size.
((OPTION_D))	In both the statement 6 specifies array size.
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	The statement 'B' is correct, because int num[6]; specifies the size of array and num[6]=21; designates the particular element(7^{th}element) of the array.

((MARKS)) (1/2/3...)	1
((QUESTION))	What does the following declaration mean? int (*ptr)[10];

((OPTION_A))	ptr is array of pointers to 10 integers
((OPTION_B))	ptr is a pointer to an array of 10 integers
((OPTION_C))	ptr is an array of 10 integers
((OPTION_D))	ptr is an pointer to array
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following statements are correct about an array? 1. The array int num[26]; can store 26 elements. 2. The expression num[1] designates the very first element in the array. 3. It is necessary to initialize the array at the time of declaration. 4. The declaration num[SIZE] is allowed if SIZE is a macro.
((OPTION_A))	1
((OPTION_B))	1,4
((OPTION_C))	2,3
((OPTION_D))	2,4
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	1. The array int num[26]; can store 26 elements. This statement is true. 2. The expression num[1] designates the very first element in the array. This statement is false, because it designates the second element of the array.

	3. It is necessary to initialize the array at the time of declaration. This statement is false. 4. The declaration num[SIZE] is allowed if SIZE is a macro. This statement is true, because the MACRO just replaces the symbol SIZE with given value. Hence the statements '1' and '4' are correct statements.

((MARKS)) (1/2/3...)	1
((QUESTION))	The smallest element of an array index called……
((OPTION_A))	Lower bound
((OPTION_B))	Upper bound
((OPTION_C))	Range
((OPTION_D))	None of Above
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following function is used to find the first occurrence of a given string in another string?
((OPTION_A))	strchr()
((OPTION_B))	strrchr()
((OPTION_C))	strstr()

((OPTION_D))	strnset()
((CORRECT_CHOICE)) (A/B/C/D)	C

((EXPLANATION)) (OPTIONAL)	char *strstr(const char *s1, const char *s2);
	Return Value:
	On success, strstr returns a pointer to the element in s1 where s2 begins (points to s2 in s1).
	On error (if s2 does not occur in s1), strstr returns null.
	Example:
	#include <stdio.h>
	#include <string.h>
	int main(void)
	{
	char *str1 = "IndiaBIX", *str2 = "ia", *ptr;
	ptr = strstr(str1, str2);
	printf("The substring is: %s\n", ptr);
	return 0;
	}
	Output: The substring is: iaBIX

((MARKS)) (1/2/3...)	1
((QUESTION))	which of the following is not an asymptotic notation?
((OPTION_A))	BIG-O
((OPTION_B))	Omega
((OPTION_C))	Phi
((OPTION_D))	Theta
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Step count for the following loop is For(int i=5; i>0; i++)
((OPTION_A))	5 times
((OPTION_B))	N+1 times
((OPTION_C))	Infinite
((OPTION_D))	No execution
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Time Complexity is
((OPTION_A))	Time required for the machine to compile the program.
((OPTION_B))	Time required for the machine to execute the program.
((OPTION_C))	Time required for the machine to debug the program.
((OPTION_D))	None
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following data structure is not linear data structure?

((OPTION_A))	Array
((OPTION_B))	Linked list
((OPTION_C))	All of above
((OPTION_D))	None of above
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the followings are application of data structure?
((OPTION_A))	Facebook
((OPTION_B))	Searching
((OPTION_C))	Sorting
((OPTION_D))	All of above
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which data structure is used for implementing recursion?
((OPTION_A))	Queue
((OPTION_B))	Stack

((OPTION_C))	Array
((OPTION_D))	List
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Representation of data structure in memory is known as…….
((OPTION_A))	recursive
((OPTION_B))	Abstract Data Type
((OPTION_C))	Storage Structure
((OPTION_D))	File Structure
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	In a linked list with n nodes, the time taken to insert an element after an element pointed by some pointer is……..
((OPTION_A))	O (1)
((OPTION_B))	O(log n)
((OPTION_C))	O(n)
((OPTION_D))	O(n log n)

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

((MARKS)) (1/2/3...)	1
((QUESTION))	In a circular linked list…….
((OPTION_A))	components are all linked together in some sequential manner.
((OPTION_B))	there is no beginning and no end.
((OPTION_C))	components are arranged hierarchically.
((OPTION_D))	forward and backward traversal within the list is permitted.
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following operators takes only integer operands?
((OPTION_A))	+
((OPTION_B))	/
((OPTION_C))	%
((OPTION_D))	*
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION))	

(OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	NULL pointer is used to define .....
((OPTION_A))	End of the linked list
((OPTION_B))	Empty list
((OPTION_C))	Empty pointer field of the structure
((OPTION_D))	All of above
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The function that return memory to heap is called.....
((OPTION_A))	Allloc()
((OPTION_B))	Malloc()
((OPTION_C))	Calloc()
((OPTION_D))	Free()
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	

((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

(A/B/C/D)	
((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 (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)	

((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))	$T(n)=\Omega(n^3)$
((OPTION_C))	$T(n)=\Omega(n^2)$
((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))	$\frac{1}{2}(n^2+n)$
((OPTION_B))	$\frac{1}{2}(n^2+3n)$
((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))	The postfix equivalent of the prefix *+ab-cd is _____.
((OPTION_A))	ab+cd-*
((OPTION_B))	ab+cd*-
((OPTION_C))	abcd+*-
((OPTION_D))	ab-cd+*
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

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

((QUESTION))	What does the following function check for? (all necessary headers to be included and function is called from main)
	
	`#define MAX 10`
	
	`typedef struct stack`
	`{`
	`int top;`
	`int item[MAX];`
	`}stack;`
	
	`int function(stack *s)`
	`{`
	`if(s->top == -1)`
	`return 1;`
	`else return 0;`
	`}`

((OPTION_A))	full stack
((OPTION_B))	invalid index
((OPTION_C))	empty stack
((OPTION_D))	infinite stack

((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	Answer: c Explanation: An empty stack is represented with the top-of-the-stack(‘top’ in this case) to be equal to -1.

MCQs

1. Which of these best describes an array?
a) A data structure that shows a hierarchical behavior
b) Container of objects of similar types
c) Container of objects of mixed types
d) All of the mentioned
Answer: b
Explanation: Array contains elements only of the same type.

2. How do you initialize an array in C?
a) int arr[3] = (1,2,3);
b) int arr(3) = {1,2,3};
c) int arr[3] = {1,2,3};
d) int arr(3) = (1,2,3);
Answer: c
Explanation: This is the syntax to initialize an array in C.

3. How do you instantiate an array in Java?
a) int arr[] = new int(3);
b) int arr[];
c) int arr[] = new int[3];
d) int arr() = new int(3);
Answer: c
Explanation: Note that option b is declaration whereas option c is to instantiate an array.

4. Which of the following is a correct way to declare a multidimensional array in Java?
a) int[][] arr;
b) int arr[][];
c) int []arr[];
d) All of the mentioned
Answer: d
Explanation: All the options are syntactically correct.

5. What is the output of the following piece of code?

```
public class array
{
        public static void main(String args[])
        {
                int []arr = {1,2,3,4,5};
                System.out.println(arr[2]);
                System.out.println(arr[4]);
        }
}
```

a) 3 and 5
b) 5 and 3

c) 2 and 4
d) 4 and 2
Answer: a
Explanation: Array indexing starts from 0.

6. What is the output of the following piece of code?

```
public class array
{
        public static void main(String args[])
        {
                int []arr = {1,2,3,4,5};
                System.out.println(arr[5]);
        }
}
```

a) 4
b) 5
c) ArrayIndexOutOfBoundsException
d) InavlidInputException
Answer: c
Explanation: Trying to access an element beyond the limits of an array gives ArrayIndexOutOfBoundsException.

7. When does the ArrayIndexOutOfBoundsException occur?
a) Compile-time
b) Run-time
c) Not an error
d) None of the mentioned
Answer: b
Explanation: ArrayIndexOutOfBoundsException is a run-time exception and the compilation is error-free.

8. Which of the following concepts make extensive use of arrays?
a) Binary trees
b) Scheduling of processes
c) Caching
d) Spatial locality
Answer: d
Explanation: Whenever a particular memory location is referred, it is likely that the locations nearby are also referred, arrays are stored as contigous blocks in memory, so if you want to access array elements, spatial locality makes it to access quickly.

9. What are the advantages of arrays?
a) Easier to store elements of same data type
b) Used to implement other data structures like stack and queue
c) Convenient way to represent matrices as a 2D array
d) All of the mentioned

Answer: d
Explanation: Arrays are simple to implement when it comes to matrices of fixed size and type, or to implement other data structures.

10. What are the disadvantages of arrays?
a) We must know before hand how many elements will be there in the array
b) There are chances of wastage of memory space if elements inserted in an array are lesser than than the allocated size
c) Insertion and deletion becomes tedious
d) All of the mentioned
Answer: d
Explanation: Arrays are of fixed size, hence during the compile time we should know its size and type, since arrays are stored in contigous locations, insertion and deletion becomes time consuming.

11. Assuming int is of 4bytes, what is the size of int arr[15];?
a) 15
b) 19
c) 11
d) 60
Answer: d
Explanation: Since there are 15 int elements and each int is of 4bytes, we get 15*4 = 60bytes.

Queues:

1. A linear list of elements in which deletion can be done from one end (front) and insertion can take place only at the other end (rear) is known as a ?
a) Queue
b) Stack
c) Tree
d) Linked list
Answer: a
Explanation: Self Explanatory.

3. A queue is a ?
a) FIFO (First In First Out) list
b) LIFO (Last In First Out) list
c) Ordered array
d) Linear tree
View Answer

Answer: a
Explanation: Self Explanatory.

4. In Breadth First Search of Graph, which of the following data structure is used?
a) Stack

b) Queue
c) Linked list
d) None of the mentioned
View Answer

Answer: b
Explanation: Self Explanatory.

5. If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time, in what order will they be removed?
a) ABCD
b) DCBA
c) DCAB
d) ABCD
Answer: a
Explanation: Queue follows FIFO approach.

6. A data structure in which elements can be inserted or deleted at/from both the ends but not in the middle is?
a) Queue
b) Circular queue
c) Dequeue
d) Priority queue
Answer: c
Explanation: Self Explanatory.

7. A normal queue, if implemented using an array of size MAX_SIZE, gets full when
a) Rear = MAX_SIZE – 1
b) Front = (rear + 1)mod MAX_SIZE
c) Front = rear + 1
d) Rear = front
Answer: a
Explanation: Condition for size of queue.

8. Queues serve major role in
a) Simulation of recursion
b) Simulation of arbitrary linked list
c) Simulation of limited resource allocation
d) All of the mentioned
Answer: c
Explanation: Rest all are implemented using other data structures.

9. Which of the following is not the type of queue?
a) Ordinary queue
b) Single ended queue
c) Circular queue

d) Priority queue
Answer: b
Explanation: Queue always has two ends.

10. In linked list implementation of queue, if only front pointer is maintained, which of the following operation take worst case linear time?
a) Insertion
b) Deletion
c) To empty a queue
d) Both a and c
Answer: d
Explanation: Since front pointer is used for deletion, so worst time for the other two cases.

11. In linked list implementation of a queue, where does a new element be inserted?
a) At the head of link list
b) At the centre position in the link list
c) At the tail of the link list
d) None of the mentioned
Answer: c
Explanation: Since queue follows FIFO so new element inserted at last.

12. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a NONEMPTY queue?
a) Only front pointer
b) Only rear pointer
c) Both front and rear pointer
d) None of the mentioned
Answer: b
Explanation: Since queue follows FIFO so new element inserted at last.