Dr.D.Y.PatilInstituteofTechnology,Pimpri,
Department of Computer Engineering

Subject Name: Principles of Programming Languages

Unit II MCQ's

1. Which of the following data types are available in Java?
   A. Primitive Datatypes
   B. Reference Datatypes
   C. Both
   D. None of the mentioned
   ANSWER: B

2. Which of the following data types comes under 'Integer' data type?
   A. short
   B. long
   C. int
   D. all of the mentioned
   ANSWER: D

3. What will be the return type of a method that not returns any value?
   A. void
   B. int
   C. double
   D. none
   ANSWER: D

4. Which keyword is used by method to refer to the object that invoked it?
   A. import
   B. catch
   C. abstract
   D. this
   ANSWER: D

5. Which of the following is a method having same name as that of its class?
   A. finalize
   B. delete
   C. class
   D. constructor
   ANSWER: D

6. Let A = {1, 2, 3}, B = {3, 4} C = {4, 5, 6} Find A x (B ? C)
   A. {(1, 2) (2, 2) (3, 4)}
   B. {(1, 4) (2, 4) (3, 4)}
   C. {(1, 4) (2, 4) (3, 4)}
   D. {(1, 4) (2, 1) (3, 4)}
   ANSWER: B

7. In Java,--is a class considered without regard to its implementation

A. Primitive Datatypes
B. Reference Datatypes
C. Abstract Data Type
D. none
ANSWER: C

8. The C language is :
A. A context free language
B. A context sensitive language
C. A regular language
D. Parsable fully only by a Turing machine
ANSWER: A

9. In an expression involving || operator, evaluation

I. Will be stopped if one of its components evaluates to false
II. Will be stopped if one of its components evaluates to true
III. Takes place from right to left
IV. Takes place from left to right

A. I and II
B. II and III
C. II and IV
D. I and III
ANSWER: C

10. Arrays are best data structures
A. for relatively permanent collections of data
B. for the size of the structure and the data in the structure are constantly changing
C. for both of above situation
D. for none of above situation
ANSWER: A

11. Two main measures for the efficiency of an algorithm are
A. Processor and memory
B. Complexity and capacity
C. Time and space
D. Data and space
ANSWER: C

12. Which of the following data structure is linear data structure?
A. Trees
B. Graphs
C. Arrays
D. None of above
ANSWER: C

13. Looping in a program means
A. Jumping to the specified branch of program
B. Repeat the specified lines of code
C. Both of above
D. None of above

ANSWER: B

14. Which of the following is selection statement in C++?
A. break
B. goto
C. exit
D. Switch
ANSWER: D

15. Break statement is used for
A. Quit a program
B. Quit the current iteration
C. Both of above
D. None of above
ANSWER: B

16. If default constructor is not defined, then how the objects of the class will be created?
A. The compiler will generate error
B. Error will occur at run-time.
C. Compiler provides its default constructor to build the object.
D. None of these
ANSWER: C

17. Assume class TEST. Which of the following statements is/are responsible to invoke copy constructor?
A. TEST T2(T1)
B. TEST T4 = T1
C. T2 = T1
D. both a and b
ANSWER: D

18. Exception generated in try block is caught in .... block.
A. catch
B. throw
C. throws
D. finally
ANSWER: A

19. Which keyword is used to explicitly throw an exception?
A. catch
B. throw
C. finally
D. throws
ANSWER: B

20. Which is a general strategy for controling how a computation evolves?
A. Recursion
B. Mapping
C. Iteration
D. None of the mentioned

ANSWER: C

21. Which of the following data types are available in Java? (GATE CS 2000)
    a) Primitive Datatypes
    b) Reference Datatypes
    c) Both
    d) None of the above

ANSWER: C

22. The format identifier '%i' in C is also used fordata type?
    a) char
    b) Int
    c) Float
    d) Double

ANSWER: B

23. Which is correct with respect to size of the datatypes?
    a) char > int > float
    b) int > char > float
    c) char < int < double
    d) double > char > int

ANSWER: C

24. Which of the following is data type stores longest floating point number.
    a) Float
    b) Double
    c) Boolean
    d) Long

ANSWER: B

25. ________________constructs the aggregates of similar type of elements
    a) Vector
    b) Array
    c) Struct
    d) Union

ANSWER: B

26. Pick the odd one out
    a) array type
    b) character type
    c) boolean type
    d) integer type

ANSWER: A

27. Pick the odd one out.

a) integer, character, boolean, floating
b) enumeration, subrange
c) integer, enum, void
d) arrays, pointer, classes

ANSWER: C

28. To perform dynamic type checking operation, the checking is done before program execution

a) True
b) False

ANSWER:B

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is not valid variable name?
((OPTION_A))	int $index;
((OPTION_B))	int index1;
((OPTION_C))	int ind1x;
((OPTION_D))	int index;
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	The variable name should not begin with special symbol like @, #, $, % and so on

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

((QUESTION))	Which of the following is true for variable names in C
((OPTION_A))	They can contain alphanumeric characters as well as special characters
((OPTION_B))	Variable names cannot start with a digit
((OPTION_C))	It is not an error to declare a variable to be one of the keywords (like goto, static)
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is true of I-value and r-values?
((OPTION_A))	An I-value is a logical value and an r-value is a real value
((OPTION_B))	I-Value are always to the left of r-value
((OPTION_C))	An I-value refers to a variable location while an r-value to its current value
((OPTION_D))	I-value are local and r-value are relative

((CORRECT _CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

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

((QUESTION))	What will be the output of the following Java program?
	class Output
	{
	        public static void main(String args[])
	        {
	           final int a=10,b=20;
	          while(a<b)
	          {
	
	          System.out.println("Hello");
	          }
	          System.out.println("World");
	
	        }
	}

((OPTION_ A))	Hello
((OPTION_ B))	run time error
((OPTION_ C))	Hello world
((OPTION_ D))	compile time error
((CORRECT _CHOICE)) (A/B/C/D)	D

((EXPLANATION)) (OPTIONAL)	Every final variable is compile time constant.

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

((QUESTION))	What will be the output of the following Java program?
	class comma_operator
	    {
	        public static void main(String args[])
	        {
	            int sum = 0;
	            for (int i = 0, j = 0; i < 5 & j < 5; ++i, j = i + 1)
	                sum += i;
	
	System.out.println(sum);
	        }
	    }

((OPTION_A))	5
((OPTION_B))	6
((OPTION_C))	14
((OPTION_D))	Compilation error
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	Using comma operator, we can include more than one statement in the initialization and iteration portion of the for loop. Therefore both ++i and j = i + 1 is executed i gets the value – 0,1,2,3,4 & j gets the values -0,1,2,3,4,5.

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

((QUESTION))	Which of this statement is incorrect?
((OPTION_A))	switch statement is more efficient than a set of nested ifs
((OPTION_B))	two case constants in the same switch can have identical values
((OPTION_C))	switch statement can only test for equality, whereas if statement can evaluate any type of boolean expression
((OPTION_D))	it is possible to create a nested switch statements
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	No two case constants in the same switch can have identical values.

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

((QUESTION))	Which is the longest scope in the following code?
	#include<stdio.h>
	int x;
	int main()
	{
	    int y;
	    fun();
	    return 0;
	}
	Void fun()
	{

	int z;
	}

((OPTION_ A))	x
((OPTION_ B))	y
((OPTION_ C))	z
((OPTION_ D))	Both a & b
((CORRECT _CHOICE)) (A/B/C/D)	A
((EXPLANA TION)) (OPTIONAL)	The variable is a global variable and its scope is entire program including all the function

((MARKS)) (1/2/3...)	1
((QUESTION))	The variable which can be accessed by all modules in a program , are called --------
((OPTION_ A))	Local variables
((OPTION_ B))	Internal variables
((OPTION_ C))	External variable
((OPTION_ D))	Global variables

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

((MARKS)) (1/2/3...)	1
((QUESTION))	The value of an automatic variable that is declared but not initialized will be -------------
((OPTION_A))	0
((OPTION_B))	1
((OPTION_C))	Unpredictable
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	If the variable is not initialized then it takes the garbage value. Hence the unpredictable value is present in the uninitialized variable

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is not data type in Pascal?

((OPTION_ A))	Real
((OPTION_ B))	Float
((OPTION_ C))	Char
((OPTION_ D))	Integer
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	What will happen if null pointer is converted to bool-----
((OPTION_ A))	The bool value is evaluated to true
((OPTION_ B))	The bool value is evaluated to false
((OPTION_ C))	Error is raised
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	In pointer, the nonzero pointer is converted to true and zero pointer is converted to false

((EXPLANATION)) (OPTIONAL)	

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

((QUESTION))	int main(void)
	{
	Char name=*p;
	‘p’=10 /* assigning to character P number 10*/
	Return 0;
	}

((OPTION_A))	Name will contain value 10
((OPTION_B))	P will contain 10
((OPTION_C))	Syntax error
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	The assignment ‘p’=10 will cause the error; I-value required as left operand of assignment

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

((QUESTION))	Choose the correct statement
((OPTION_A))	Reference is stored on stack
((OPTION_B))	Reference is stored on heap
((OPTION_C))	Reference is stored on queue
((OPTION_D))	None of these
((CORRECT_CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Discriminated unions are supported by-----
((OPTION_A))	C
((OPTION_B))	C++
((OPTION_C))	ML
((OPTION_D))	All of these

((CORRECT _CHOICE)) (A/B/C/D)	C
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	In C, reference is declared using symbol-----
((OPTION_A))	*
((OPTION_B))	&
((OPTION_C))	&&
((OPTION_D))	!
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Chose the correct statement I. Array of references can be created II. Change in reference changes the referent

((OPTION_ A))	Only I is correct
((OPTION_ B))	Only II is correct
((OPTION_ C))	Both I and II is correct
((OPTION_ D))	Neither I and II is correct
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Choose the correct statement
((OPTION_ A))	Reference must be initialized within a function
((OPTION_ B))	Reference must be initialized outsidea function
((OPTION_ C))	Reference must be always initialized
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	C

((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Choose the correct statement
((OPTION_ A))	A reference is not a constant pointer
((OPTION_ B))	Reference is automatically de-referenced
((OPTION_ C))	Both a & b
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Choose the correct statement
((OPTION_ A))	A reference variable need not be de-referenced to access value

((OPTION_ B))	A reference variable need to be de-referenced to access value
((OPTION_ C))	It depends upon the type of reference whether to de-refer it to access value
((OPTION_ D))	None of these
((CORRECT _CHOICE))	A
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Choose the correct statement I The variable and its reference are linked together II We can change the value of variable via its reference
((OPTION_ A))	Only I
((OPTION_ B))	Only II
((OPTION_ C))	Both I & II
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	C

((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	If an expression contain double, float, and long then the data type of that expression becomes-----
((OPTION_A))	Int
((OPTION_B))	Float
((OPTION_C))	Long
((OPTION_D))	Double
((CORRECT_CHOICE)) (A/B/C/D)	d
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Explicit type conversion is known as ------
((OPTION_A))	Conversion

((OPTION_ B))	Casting
((OPTION_ C))	Separation
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	What is strong type system?
((OPTION_ A))	The type system in which only built-in data types are allowed
((OPTION_ B))	The type system in which only user defined data types are allowed
((OPTION_ C))	The type system that guarantees not to generate type errors
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	C

((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	For representing logical values ----- data type is used in C++
((OPTION_ A))	int
((OPTION_ B))	bool
((OPTION_ C))	char
((OPTION_ D))	All of the above
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	The C++ allows the boolen data type for denoting true or false values

((MARKS)) (1/2/3...)	1
((QUESTION))	Looping in a program means-----
((OPTION_ A))	Jumping to the specified branch of program

((OPTION_ B))	Repeat the specified lines of code
((OPTION_ C))	Both of above
((OPTION_ D))	None of these
((CORRECT _CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	What is required to reference an element in an array?
((OPTION_ A))	Array name
((OPTION_ B))	Index value of the element
((OPTION_ C))	Element value
((OPTION_ D))	Both a and b
((CORRECT _CHOICE)) (A/B/C/D)	D

((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The following the exit controlled loop-------------
((OPTION_A))	While
((OPTION_B))	do-while
((OPTION_C))	for
((OPTION_D))	go to
((CORRECT_CHOICE)) (A/B/C/D)	B
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The scope of a variable refer to-----
((OPTION_A))	The length of the variable

((OPTION_ B))	The name of the variable
((OPTION_ C))	The accessibility of the variable
((OPTION_ D))	The lifetime of the variable
((CORRECT _CHOICE)) (A/B/C/D)	C
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	The goal of structured programming is to---
((OPTION_ A))	Have well indented programs
((OPTION_ B))	Be able to infer the flow of control from the compiled code
((OPTION_ C))	Be able to infer the flow of control from the program text
((OPTION_ D))	Avoid the use of GOTO statements
((CORRECT _CHOICE)) (A/B/C/D)	C
((EXPLANA TION)) (OPTIONAL)	

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

((QUESTION))	Functions and subprograms are both same
((OPTION_A))	True
((OPTION_B))	False
((OPTION_C))	
((OPTION_D))	
((CORRECT_CHOICE)) (A/B/C/D)	b
((EXPLANATION)) (OPTIONAL)	A subprogram consists of procedures and functions. Both of them are collectively called subprograms. So, subprogram is not same as a function but a function is a part of subprogram in case of VHDL.

((MARKS)) (1/2/3...)	1
((QUESTION))	The function is called from the ________
((OPTION_A))	Function itself
((OPTION_B))	Library
((OPTION_C))	Main code
((OPTION_D))	Package
((CORRECT_CHOICE)) (A/B/C/D)	c
((EXPLANATION)) (OPTIONAL)	The function which is once declared is always called from the main code. Whenever a function call occurs, the control is passed to the space where the function is defined. Then, the function is executed till a RETURN statement comes, which returns the control to main code.

((MARKS)) (1/2/3...)	1
((QUESTION))	Which among the following best defines abstraction?
((OPTION_A))	Hiding the implementation
((OPTION_B))	Showing the important data
((OPTION_C))	Hiding the important data
((OPTION_D))	Hiding the implementation and showing only the features
((CORRECT_CHOICE)) (A/B/C/D)	d
((EXPLANATION)) (OPTIONAL)	It includes hiding the implementation part and showing only the required data and features to the user. It is done to hide the implementation complexity and details from the user. And to provide a good interface in programming.

((MARKS)) (1/2/3...)	1
((QUESTION))	Abstraction can apply to _____________
((OPTION_A))	Control and data
((OPTION_B))	Only data
((OPTION_C))	Only control
((OPTION_D))	Classes
((CORRECT_CHOICE)) (A/B/C/D)	a

((EXPLANATION)) (OPTIONAL)	Abstraction applies to both. Control abstraction involves use of subroutines and control flow abstraction. Data abstraction involves handling pieces of data in meaningful ways.

((MARKS)) (1/2/3...)	1
((QUESTION))	When there is multiple IF and a single ELSE then the ELSE part doesn't get a clear view to go with which IF, this problem is called
((OPTION_A))	dangling if problem
((OPTION_B))	Multiple if problem
((OPTION_C))	dangling else problem
((OPTION_D))	None of the above
((CORRECT_CHOICE)) (A/B/C/D)	c
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	How is iteration controlled?
((OPTION_A))	Logical
((OPTION_B))	counter
((OPTION_C))	Both a and b

((OPTION_ D))	None of the above
((CORRECT _CHOICE)) (A/B/C/D)	c
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	In Logically-Controlled Loops repetition control is based on
((OPTION_ A))	counter
((OPTION_ B))	Boolenexperssion
((OPTION_ C))	combination of the A & B
((OPTION_ D))	None of the above
((CORRECT _CHOICE)) (A/B/C/D)	b
((EXPLANA TION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Relational operator always have lower precedence than the arithmetic operators
((OPTION_ A))	true

((OPTION_ B))	false
((OPTION_ C))	
((OPTION_ D))	
((CORRECT _CHOICE)) (A/B/C/D)	a
((EXPLANA TION)) (OPTIONAL)	

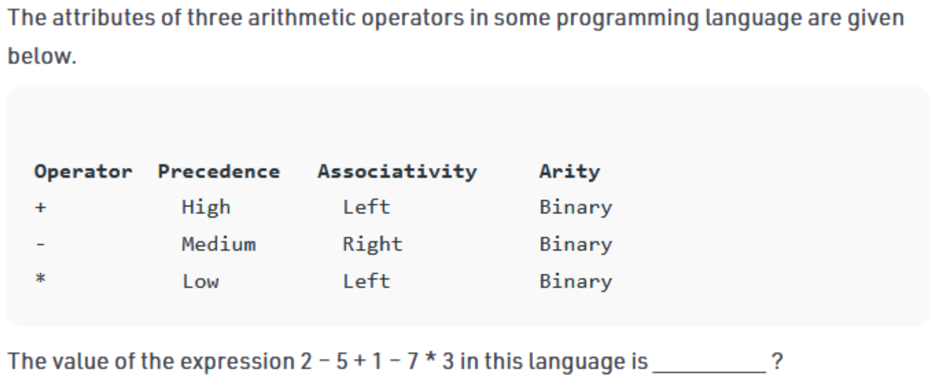
((MARKS)) (1/2/3...)	1
((QUESTION))	In which type subscript ranges are statically bound, but the allocation is done at declaration elaboration time during execution
((OPTION_ A))	Fixed stack-dynamic array
((OPTION_ B))	Stack-dynamic array
((OPTION_ C))	Fixed heap dynamic array
((OPTION_ D))	Heap-dynamic array
((CORRECT _CHOICE)) (A/B/C/D)	a
((EXPLANA TION)) (OPTIONAL)	

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

((QUESTION))	Binding of subscript ranges and storage allocation is dynamic and can change any number of times
((OPTION_A))	Fixed heap dynamic array
((OPTION_B))	Heap-dynamic array
((OPTION_C))	Stack-dynamic array
((OPTION_D))	Static array
((CORRECT_CHOICE)) (A/B/C/D)	b
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	An unordered collection of data elements that are indexed by an equal number of values called keys are called ---------
((OPTION_A))	rectangular array
((OPTION_B))	Jagged Arrays
((OPTION_C))	record
((OPTION_D))	Associative Arrays
((CORRECT_CHOICE)) (A/B/C/D)	d
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	Heterogeneous Arrays supported by which language
((OPTION_A))	C
((OPTION_B))	Ada
((OPTION_C))	Python
((OPTION_D))	ML
((CORRECT_CHOICE)) (A/B/C/D)	c
((EXPLANATION)) (OPTIONAL)	

((MARKS)) (1/2/3...)	1
((QUESTION))	 The attributes of three arithmetic operators in some programming language are given below. Operator Precedence Associativity Arity + High Left Binary - Medium Right Binary * Low Left Binary The value of the expression 2 – 5 + 1 – 7 * 3 in this language is __________ ?
((OPTION_A))	1
((OPTION_B))	2
((OPTION_C))	3

((OPTION_ D))	9
((CORRECT _CHOICE)) (A/B/C/D)	D
((EXPLANA TION)) (OPTIONAL)	

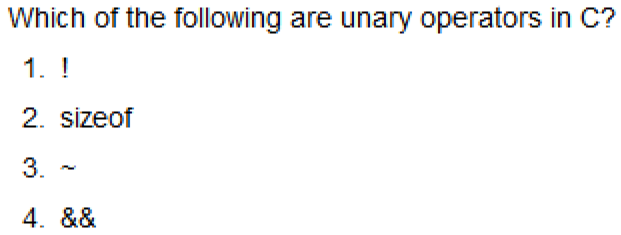
((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is the correct order of evaluation for the below expression? *z = x + y * z / 4 % 2 - 1*
((OPTION_ A))	* / % + - =
((OPTION_ B))	= * / % + -
((OPTION_ C))	/ * % - + =
((OPTION_ D))	* % / - + =
((CORRECT _CHOICE)) (A/B/C/D)	A
((EXPLANA TION)) (OPTIONAL)	C uses left associativity for evaluating expressions to break a tie between two operators having same precedence

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is the correct usage of conditional operators used in C?

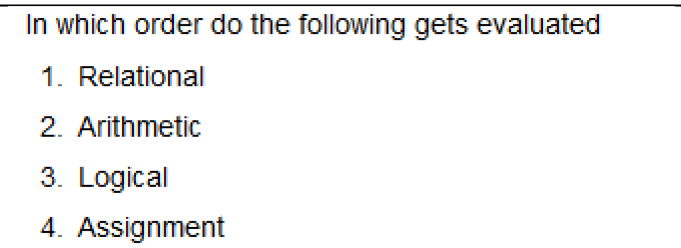
((OPTION_ A))	a>b ? c=30 : c=40;
((OPTION_ B))	a>b ? c=30;
((OPTION_ C))	max = a>b ? a>c?a:c:b>c?b:c
((OPTION_ D))	return (a>b)?(a:b)
((CORRECT _CHOICE)) (A/B/C/D)	C
((EXPLANA TION)) (OPTIONAL)	Option A: assignment statements are always return in paranthesis in the case of conditional operator. It should be a>b? (c=30):(c=40); Option B: it is syntatically wrong. Option D: syntatically wrong, it should be return(a>b ? a:b); Option C: it uses nested conditional operator, this is logic for finding greatest number out of three numbers.

((MARKS)) (1/2/3...)	1
((QUESTION))	Which of the following is the correct order if calling functions in the below code? *a = f1(23, 14) * f2(12/4) + f3();*
((OPTION_ A))	f1, f2, f3
((OPTION_ B))	f3, f2, f1
((OPTION_ C))	Order may vary from compiler to compiler
((OPTION_ D))	None of above
((CORRECT _CHOICE)) (A/B/C/D)	C

((EXPLANATION)) (OPTIONAL)	Here, Multiplication will happen before the addition, but in which order the functions would be called is undefined. In an arithmetic expression the parenthesis tell the compiler which operands go with which operators but do not force the compiler to evaluate everything within the parenthesis first.

((MARKS)) (1/2/3...)	1
((QUESTION))	 Which of the following are unary operators in C? 1. ! 2. sizeof 3. ~ 4. &&
((OPTION_A))	1,2
((OPTION_B))	1,3
((OPTION_C))	2,4
((OPTION_D))	1,2,3
((CORRECT_CHOICE)) (A/B/C/D)	D
((EXPLANATION)) (OPTIONAL)	An operation with only one operand is called unary operation. **Unary operators**: *!* Logical NOT operator. ~bitwise NOT operator. *sizeof* Size-of operator. *&&* Logical AND is a logical operator. Therefore, 1, 2, 3 are unary operators.

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

((QUESTION))	 In which order do the following gets evaluated 1. Relational 2. Arithmetic 3. Logical 4. Assignment
((OPTION_ A))	2134
((OPTION_ B))	1234
((OPTION_ C))	4321
((OPTION_ D))	3214
((CORRECT _CHOICE)) (A/B/C/D)	A
((EXPLANATION)) (OPTIONAL)	2. Arithmetic operators: *, /, %, +, - 1. Relational operators: >, <, >=, <=, ==, != 3. Logical operators :!, &&, || 4. Assignment operators: =