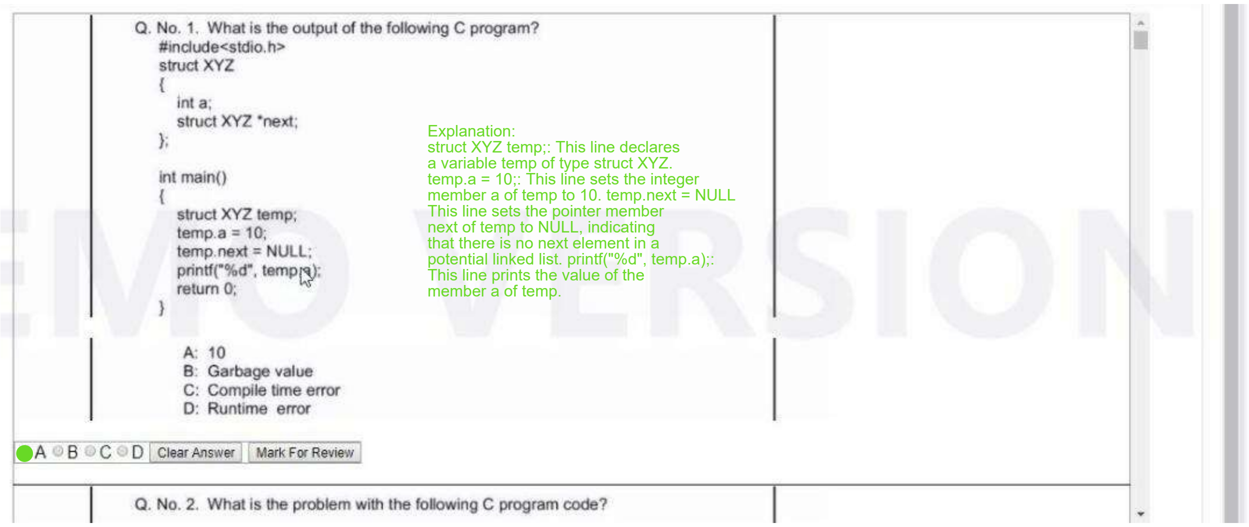


Q. No. 1. What is the output of the following C program?

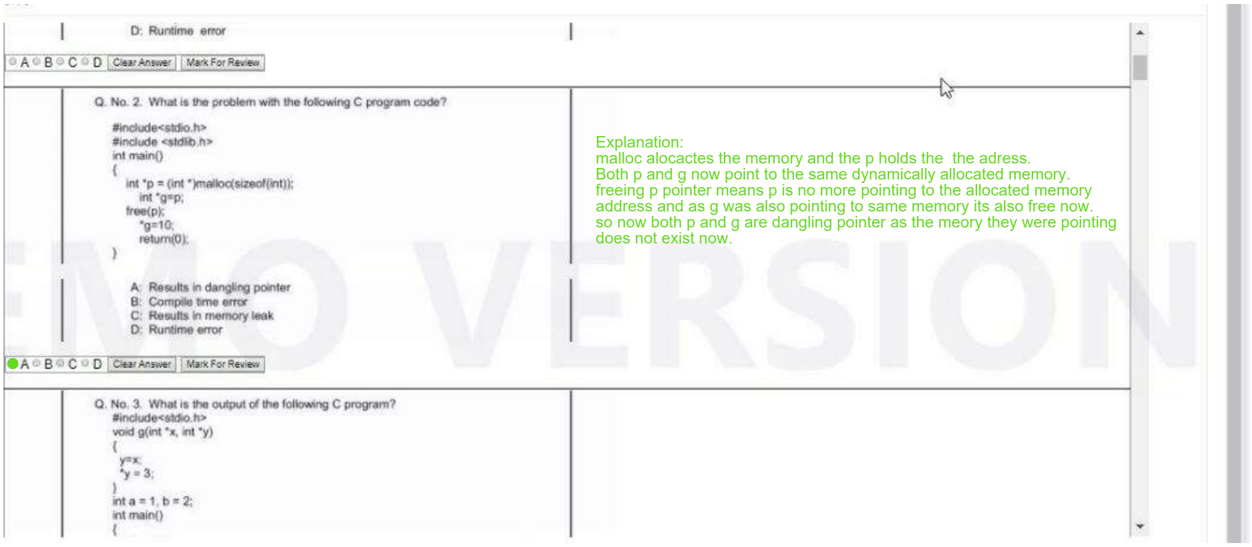
```
#include<stdio.h>
struct XYZ
{
    int a;
    struct XYZ *next;
};

int main()
{
    struct XYZ temp;
    temp.a = 10;
    temp.next = NULL;
    printf("%d", temp.a);
    return 0;
}
```

A: 10
B: Garbage value
C: Compile time error
D: Runtime error

● A ○ B ○ C ○ D
Clear Answer
Mark For Review

Q. No. 2. What is the problem with the following C program code?

Explanation:
struct XYZ temp;: This line declares a variable temp of type struct XYZ. temp.a = 10;: This line sets the integer member a of temp to 10. temp.next = NULL This line sets the pointer member next of temp to NULL, indicating that there is no next element in a potential linked list. printf("%d", temp.a);: This line prints the value of the member a of temp.



D: Runtime error

○ A ○ B ○ C ○ D
Clear Answer
Mark For Review

Q. No. 2. What is the problem with the following C program code?

```
#include<stdio.h>
#include <stdlib.h>
int main()
{
   int *p = (int *)malloc(sizeof(int));
      int *g=p;
   free(p);
      *g=10;
      return(0);
}
```

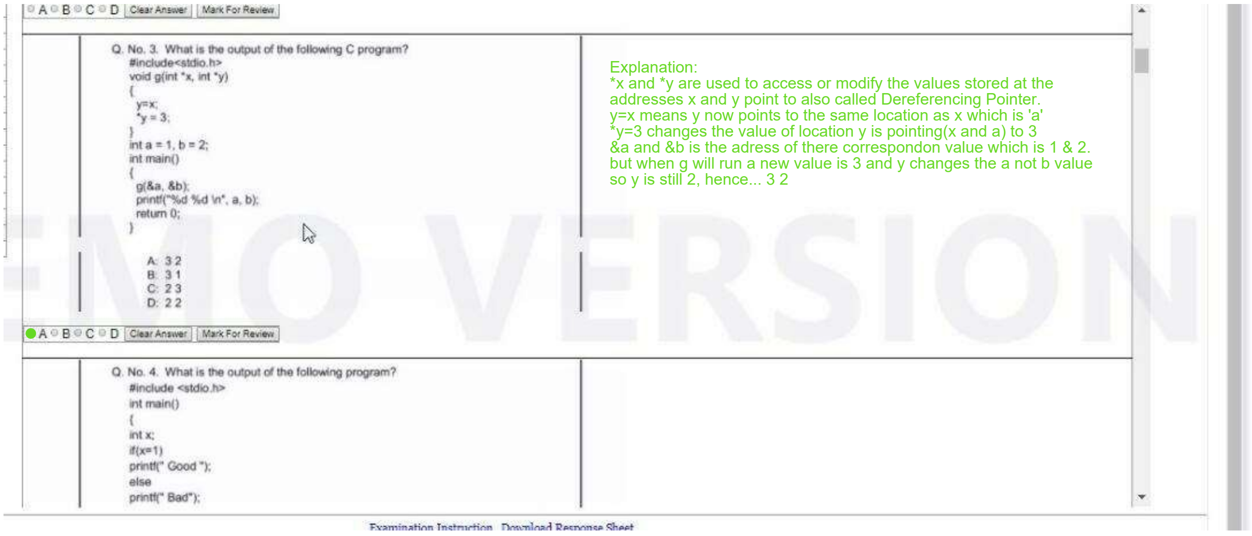
A: Results in dangling pointer
B: Compile time error
C: Results in memory leak
D: Runtime error

● A ○ B ○ C ○ D
Clear Answer
Mark For Review

Q. No. 3. What is the output of the following C program?

```
#include<stdio.h>
void g(int *x, int *y)
{
  y=x;
  *y = 3;
}
int a = 1, b = 2;
int main()
{
```

Explanation:
malloc alocactes the memory and the p holds the the adress.
Both p and g now point to the same dynamically allocated memory.
freeing p pointer means p is no more pointing to the allocated memory address and as g was also pointing to same memory its also free now.
so now both p and g are dangling pointer as the meory they were pointing does not exist now.



○ A ○ B ○ C ○ D
Clear Answer
Mark For Review

Q. No. 3. What is the output of the following C program?

```
#include<stdio.h>
void g(int *x, int *y)
{
  y=x;
  *y = 3;
}
int a = 1, b = 2;
int main()
{
  g(&a, &b);
  printf("%d %d \n", a, b);
  return 0;
}
```

A: 3 2
B: 3 1
C: 2 3
D: 2 2

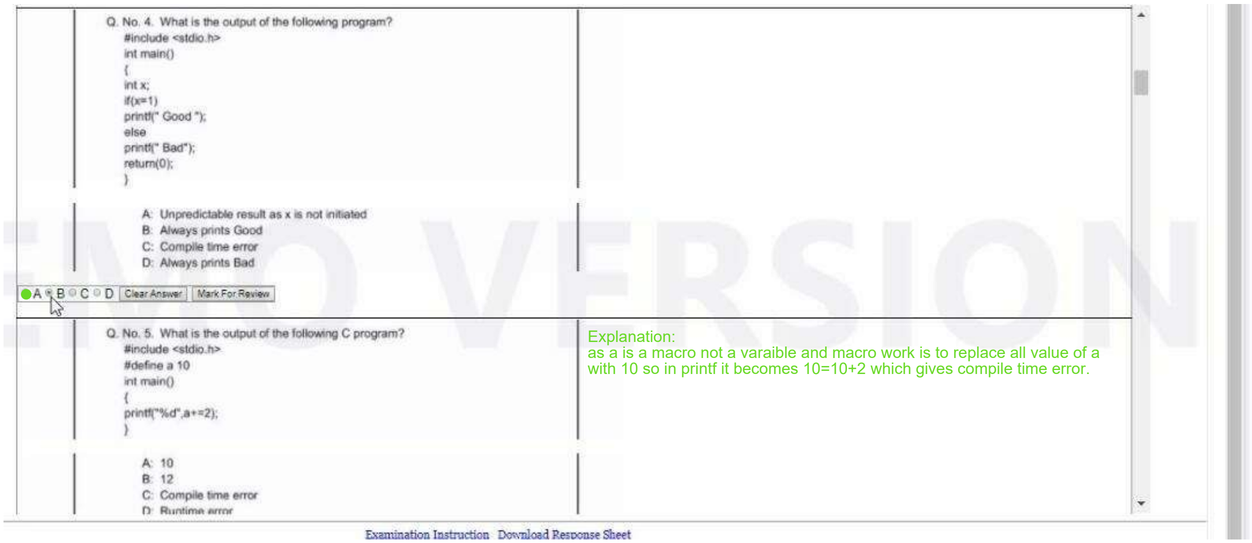
● A ○ B ○ C ○ D
Clear Answer
Mark For Review

Q. No. 4. What is the output of the following program?

```
#include <stdio.h>
int main()
{
int x;
if(x=1)
printf(" Good ");
else
printf(" Bad");
```

Examination Instruction
Download Response Sheet

Explanation:
*x and *y are used to access or modify the values stored at the addresses x and y point to also called Dereferencing Pointer.
y=x means y now points to the same location as x which is 'a'
*y=3 changes the value of location y is pointing(x and a) to 3
&a and &b is the adress of there correspondon value which is 1 & 2.
but when g will run a new value is 3 and y changes the a not b value so y is still 2, hence... 3 2



● A ○ B ○ C ○ D
Clear Answer
Mark For Review
EMO VERSION
Examination Instruction
Download Response Sheet

Q. No. 4. What is the output of the following program?

```
#include <stdio.h>
int main()
{
int x;
if(x=1)
printf(" Good ");
else
printf(" Bad");
return(0);
}
```

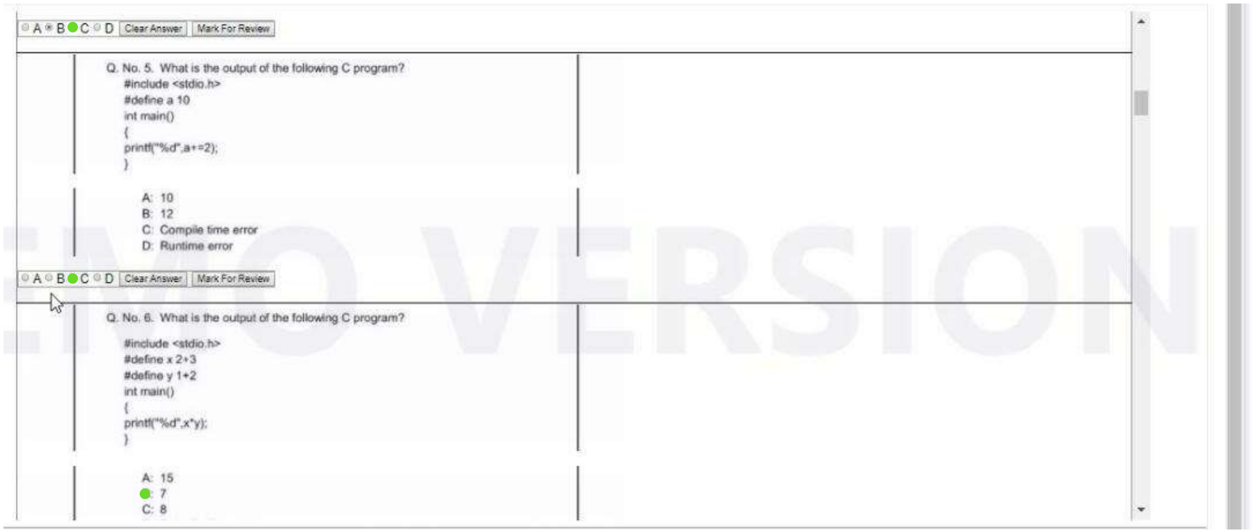
A: Unpredictable result as x is not initiated
B: Always prints Good
C: Compile time error
D: Always prints Bad

Q. No. 5. What is the output of the following C program?

```
#include <stdio.h>
#define a 10
int main()
{
printf("%d",a+=2);
}
```

A: 10
B: 12
C: Compile time error
D: Runtime error

Explanation:
as a is a macro not a varaible and macro work is to replace all value of a with 10 so in printf it becomes 10=10+2 which gives compile time error.



○ A ○ B ● C ○ D
Clear Answer
Mark For Review
EMO VERSION

Q. No. 5. What is the output of the following C program?

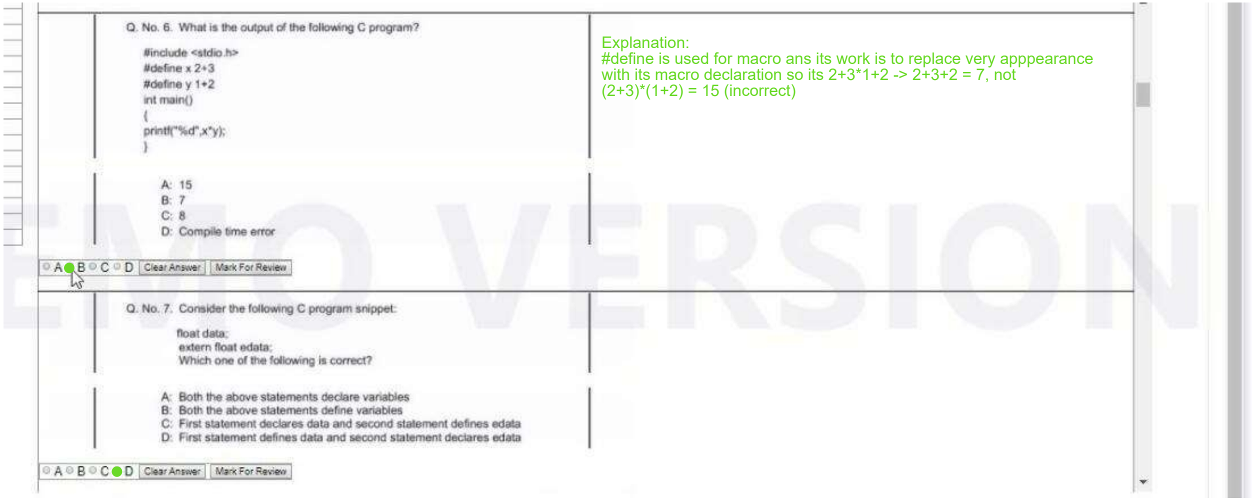
```
#include <stdio.h>
#define a 10
int main()
{
printf("%d",a+=2);
}
```

A: 10
B: 12
C: Compile time error
D: Runtime error

Q. No. 6. What is the output of the following C program?

```
#include <stdio.h>
#define x 2+3
#define y 1+2
int main()
{
printf("%d",x*y);
}
```

A: 15
●: 7
C: 8



○ A ● B ○ C ○ D
○ A ○ B ○ C ● D
Clear Answer
Mark For Review
EMO VERSION

Q. No. 6. What is the output of the following C program?

```
#include <stdio.h>
#define x 2+3
#define y 1+2
int main()
{
printf("%d",x*y);
}
```

A: 15
B: 7
C: 8
D: Compile time error

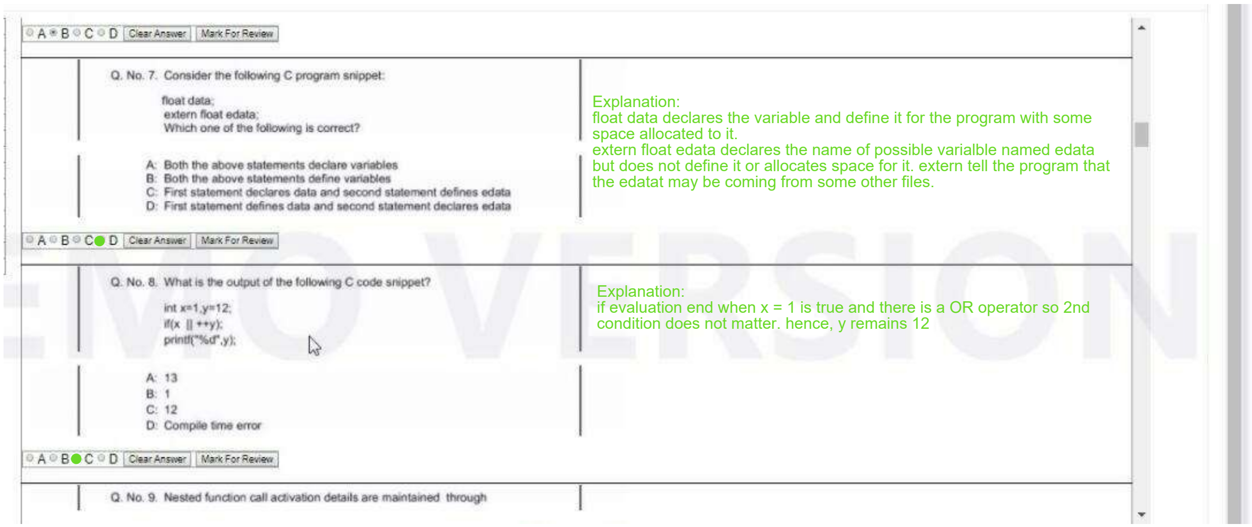
Explanation:
#define is used for macro ans its work is to replace very apppearance with its macro declaration so its 2+3*1+2 -> 2+3+2 = 7, not (2+3)*(1+2) = 15 (incorrect)

Q. No. 7. Consider the following C program snippet:

```
float data;
extern float edata;
```

Which one of the following is correct?

A: Both the above statements declare variables
B: Both the above statements define variables
C: First statement declares data and second statement defines edata
D: First statement defines data and second statement declares edata


○ A ◉ B ○ C ○ D Clear Answer Mark For Review

Q. No. 7. Consider the following C program snippet:

```
float data;
extern float edata;
```

Which one of the following is correct?

A: Both the above statements declare variables
B: Both the above statements define variables
C: First statement declares data and second statement defines edata
D: First statement defines data and second statement declares edata

Explanation:
float data declares the variable and define it for the program with some space allocated to it.
extern float edata declares the name of possible varialble named edata but does not define it or allocates space for it. extern tell the program that the edatat may be coming from some other files.

○ A ○ B ○ C ● D Clear Answer Mark For Review

Q. No. 8. What is the output of the following C code snippet?

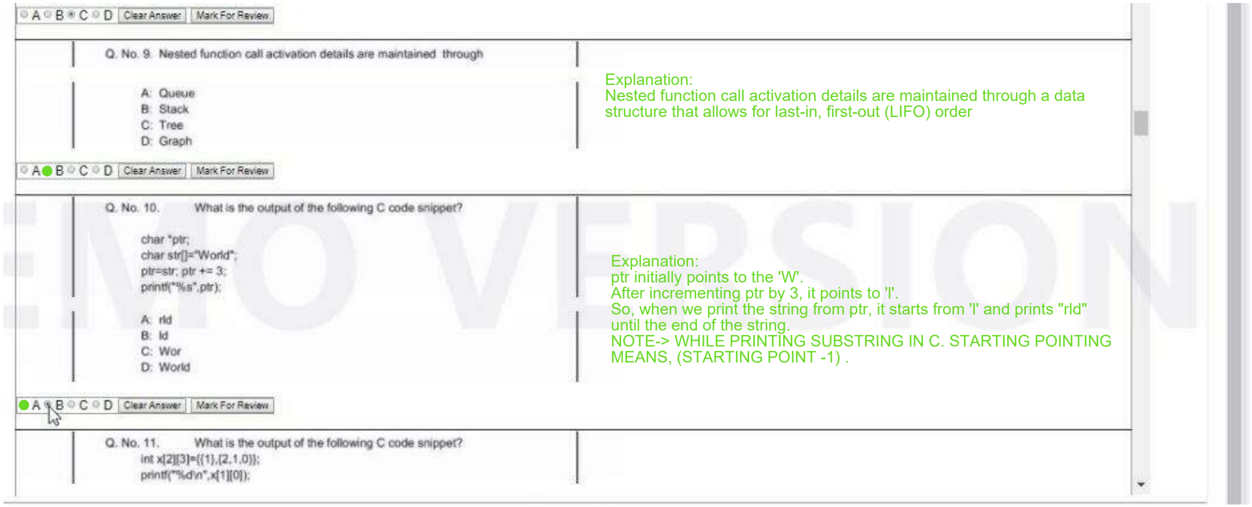
```
int x=1,y=12;
if(x || ++y);
printf("%d",y);
```

A: 13
B: 1
C: 12
D: Compile time error

Explanation:
if evaluation end when x = 1 is true and there is a OR operator so 2nd condition does not matter. hence, y remains 12

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No. 9. Nested function call activation details are maintained through


○ A ○ B ◉ C ○ D Clear Answer Mark For Review

Q. No. 9. Nested function call activation details are maintained through

A: Queue
B: Stack
C: Tree
D: Graph

Explanation:
Nested function call activation details are maintained through a data structure that allows for last-in, first-out (LIFO) order

○ A ● B ○ C ○ D Clear Answer Mark For Review

Q. No. 10. What is the output of the following C code snippet?

```
char *ptr;
char str[]="World";
ptr=str; ptr += 3;
printf("%s",ptr);
```

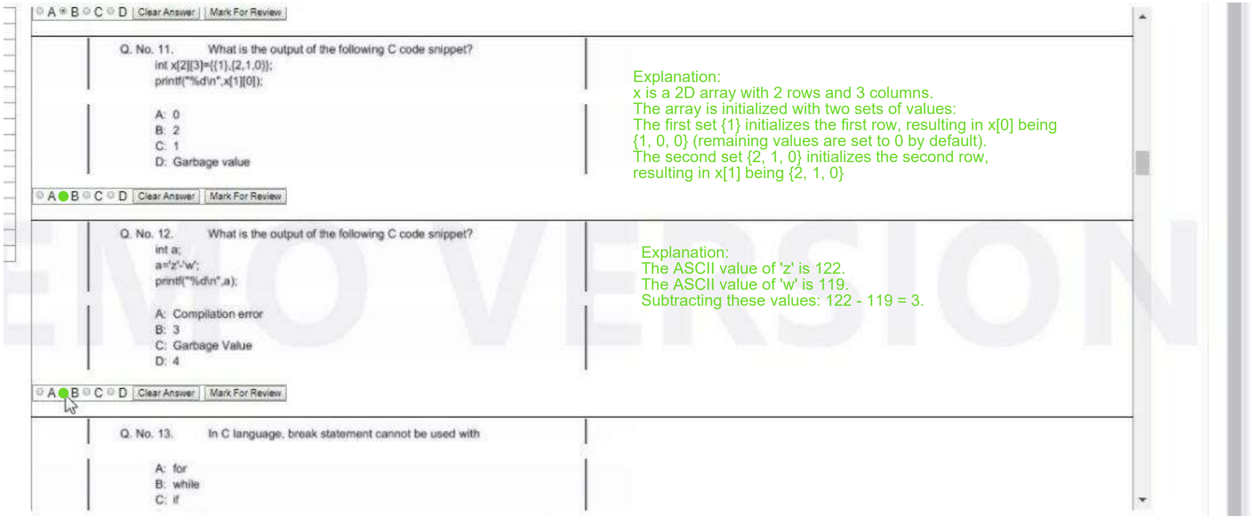
A: rld
B: ld
C: Wor
D: World

Explanation:
ptr initially points to the 'W'.
After incrementing ptr by 3, it points to 'l'.
So, when we print the string from ptr, it starts from 'l' and prints "rld" until the end of the string.
NOTE-> WHILE PRINTING SUBSTRING IN C. STARTING POINTING MEANS, (STARTING POINT -1) .

● A ○ B ○ C ○ D Clear Answer Mark For Review

Q. No. 11. What is the output of the following C code snippet?

```
int x[2][3]={{1},{2,1,0}};
printf("%d\n",x[1][0]);
```


○ A ◉ B ○ C ○ D Clear Answer Mark For Review

Q. No. 11. What is the output of the following C code snippet?

```
int x[2][3]={{1},{2,1,0}};
printf("%d\n",x[1][0]);
```

A: 0
B: 2
C: 1
D: Garbage value

Explanation:
x is a 2D array with 2 rows and 3 columns.
The array is initialized with two sets of values:
The first set {1} initializes the first row, resulting in x[0] being {1, 0, 0} (remaining values are set to 0 by default).
The second set {2, 1, 0} initializes the second row, resulting in x[1] being {2, 1, 0}

○ A ● B ○ C ○ D Clear Answer Mark For Review

Q. No. 12. What is the output of the following C code snippet?

```
int a;
a='z'-'w';
printf("%d\n",a);
```

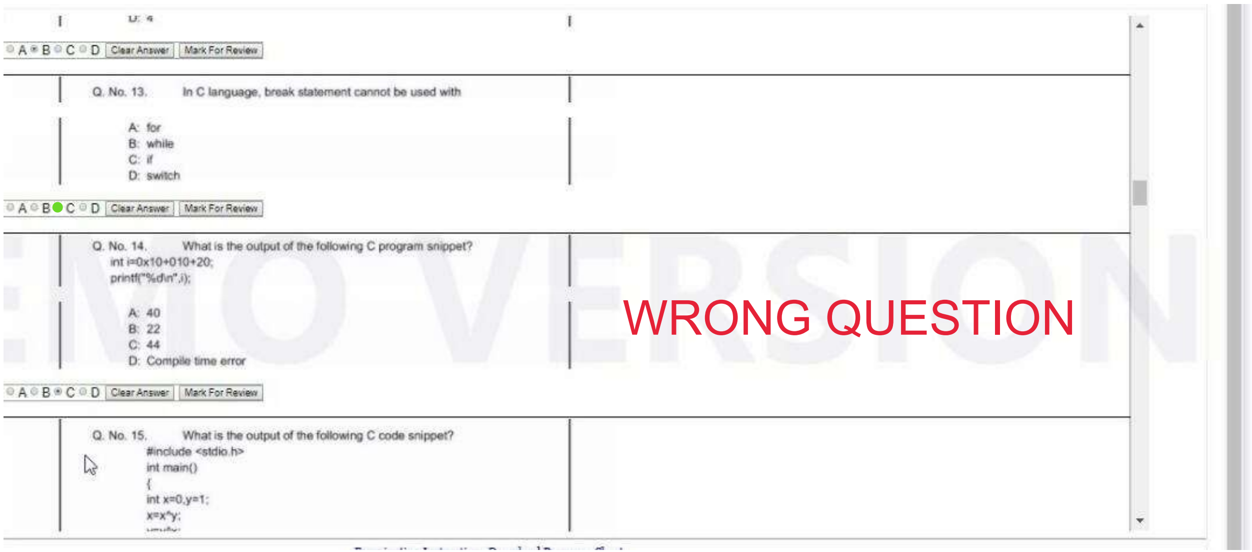
A: Compilation error
B: 3
C: Garbage Value
D: 4

Explanation:
The ASCII value of 'z' is 122.
The ASCII value of 'w' is 119.
Subtracting these values: 122 - 119 = 3.

○ A ● B ○ C ○ D Clear Answer Mark For Review

Q. No. 13. In C language, break statement cannot be used with

A: for
B: while
C: if



D: 4

○ A ◉ B ○ C ○ D Clear Answer Mark For Review

Q. No. 13. In C language, break statement cannot be used with

A: for
B: while
C: if
D: switch

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No. 14. What is the output of the following C program snippet?

```
int i=0x10+010+20;
printf("%d\n",i);
```

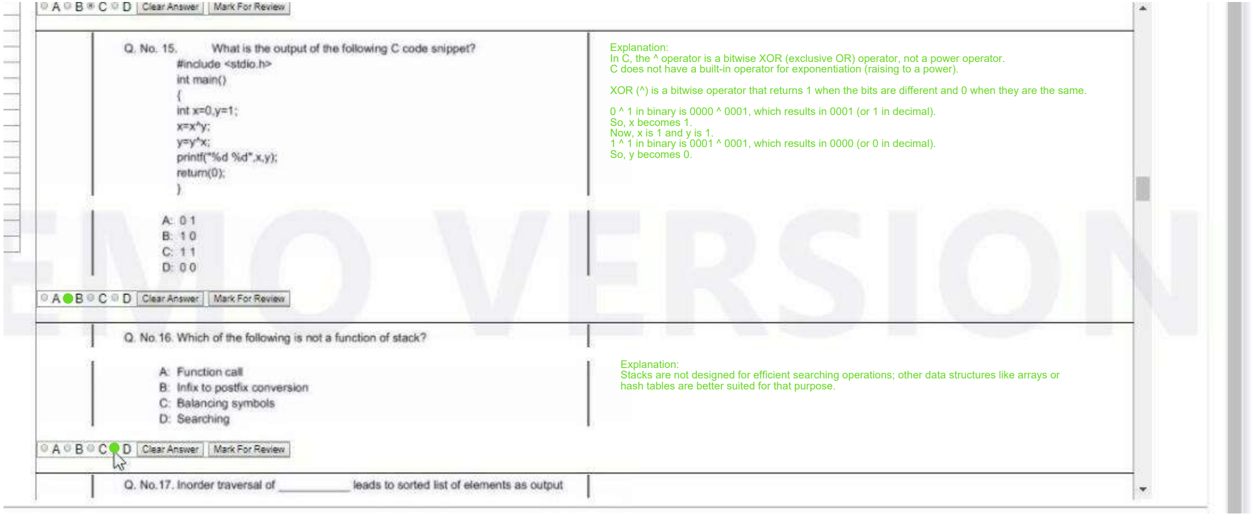
A: 40
B: 22
C: 44
D: Compile time error

WRONG QUESTION

○ A ○ B ◉ C ○ D Clear Answer Mark For Review

Q. No. 15. What is the output of the following C code snippet?

```
#include <stdio.h>
int main()
{
int x=0,y=1;
x=x^y;
```



○ A ○ B ◉ C ○ D Clear Answer Mark For Review

Q. No. 15. What is the output of the following C code snippet?

```
#include <stdio.h>
int main()
{
int x=0,y=1;
x=x^y;
y=y^x;
printf("%d %d",x,y);
return(0);
}
```

A: 0 1
B: 1 0
C: 1 1
D: 0 0

○ A ● B ○ C ○ D Clear Answer Mark For Review

Explanation:
In C, the ^ operator is a bitwise XOR (exclusive OR) operator, not a power operator.
C does not have a built-in operator for exponentiation (raising to a power).

XOR (^) is a bitwise operator that returns 1 when the bits are different and 0 when they are the same.

0 ^ 1 in binary is 0000 ^ 0001, which results in 0001 (or 1 in decimal).
So, x becomes 1.
Now, x is 1 and y is 1.
1 ^ 1 in binary is 0001 ^ 0001, which results in 0000 (or 0 in decimal).
So, y becomes 0.

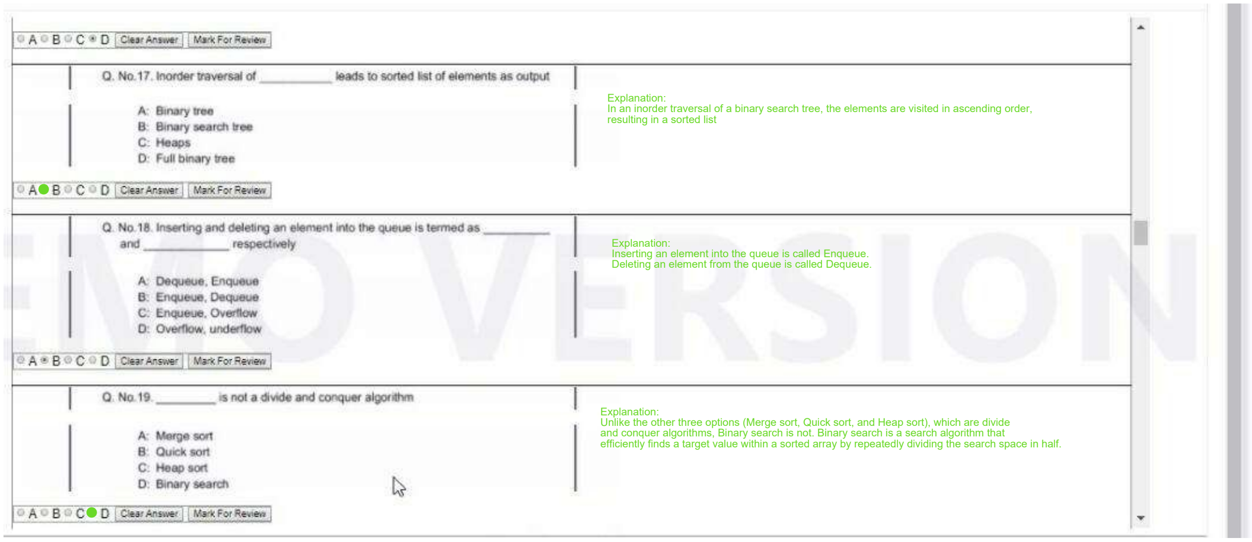
Q. No.16. Which of the following is not a function of stack?

A: Function call
B: Infix to postfix conversion
C: Balancing symbols
D: Searching

○ A ○ B ○ C ● D Clear Answer Mark For Review

Explanation:
Stacks are not designed for efficient searching operations; other data structures like arrays or hash tables are better suited for that purpose.

Q. No.17. Inorder traversal of ________ leads to sorted list of elements as output



○ A ○ B ○ C ◉ D Clear Answer Mark For Review

Q. No.17. Inorder traversal of ________ leads to sorted list of elements as output

A: Binary tree
B: Binary search tree
C: Heaps
D: Full binary tree

○ A ● B ○ C ○ D Clear Answer Mark For Review

Explanation:
In an inorder traversal of a binary search tree, the elements are visited in ascending order, resulting in a sorted list

Q. No.18. Inserting and deleting an element into the queue is termed as ________ and ________ respectively

A: Dequeue, Enqueue
B: Enqueue, Dequeue
C: Enqueue, Overflow
D: Overflow, underflow

○ A ◉ B ○ C ○ D Clear Answer Mark For Review

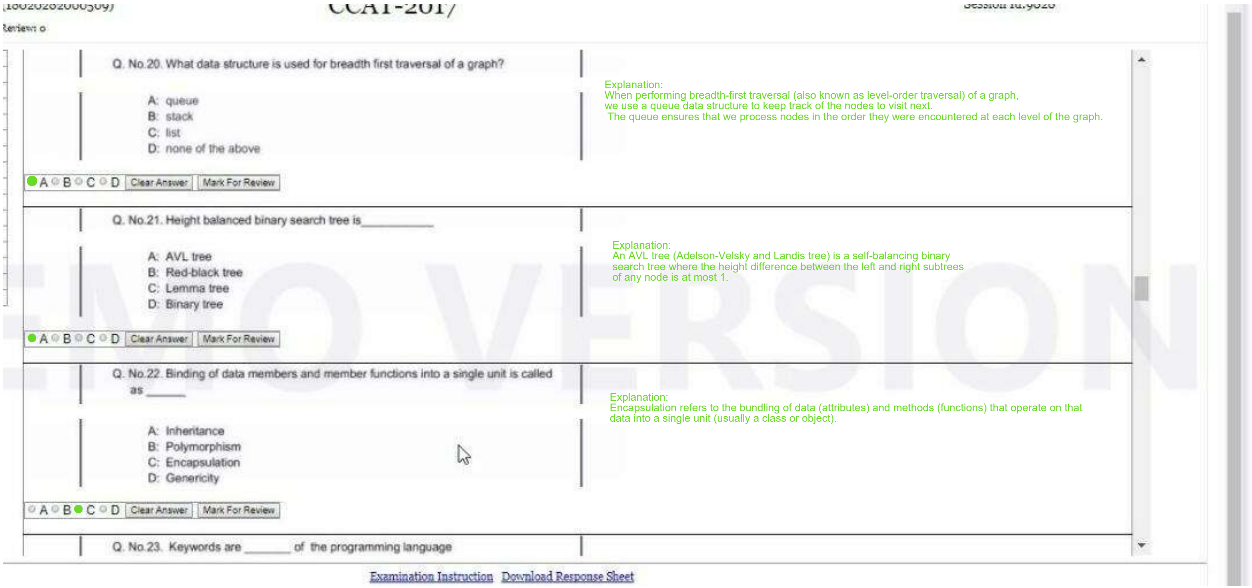
Explanation:
Inserting an element into the queue is called Enqueue.
Deleting an element from the queue is called Dequeue.

Q. No.19. ________ is not a divide and conquer algorithm

A: Merge sort
B: Quick sort
C: Heap sort
D: Binary search

○ A ○ B ○ C ● D Clear Answer Mark For Review

Explanation:
Unlike the other three options (Merge sort, Quick sort, and Heap sort), which are divide and conquer algorithms, Binary search is not. Binary search is a search algorithm that efficiently finds a target value within a sorted array by repeatedly dividing the search space in half.



Q. No.20. What data structure is used for breadth first traversal of a graph?

A: queue
B: stack
C: list
D: none of the above

● A ○ B ○ C ○ D Clear Answer Mark For Review

Explanation:
When performing breadth-first traversal (also known as level-order traversal) of a graph,
we use a queue data structure to keep track of the nodes to visit next.
The queue ensures that we process nodes in the order they were encountered at each level of the graph.

Q. No.21. Height balanced binary search tree is__________

A: AVL tree
B: Red-black tree
C: Lemma tree
D: Binary tree

● A ○ B ○ C ○ D Clear Answer Mark For Review

Explanation:
An AVL tree (Adelson-Velsky and Landis tree) is a self-balancing binary
search tree where the height difference between the left and right subtrees
of any node is at most 1.

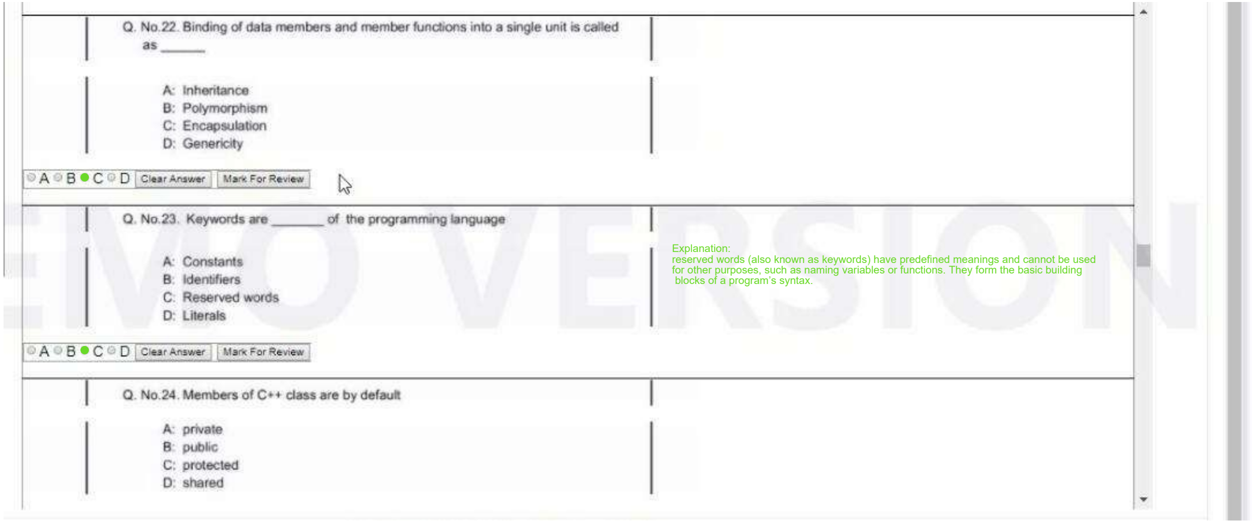
Q. No.22. Binding of data members and member functions into a single unit is called as ______

A: Inheritance
B: Polymorphism
C: Encapsulation
D: Genericity

○ A ○ B ● C ○ D Clear Answer Mark For Review

Explanation:
Encapsulation refers to the bundling of data (attributes) and methods (functions) that operate on that
data into a single unit (usually a class or object).

Q. No.23. Keywords are ______ of the programming language



Q. No.22. Binding of data members and member functions into a single unit is called as ______

A: Inheritance
B: Polymorphism
C: Encapsulation
D: Genericity

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.23. Keywords are _______ of the programming language

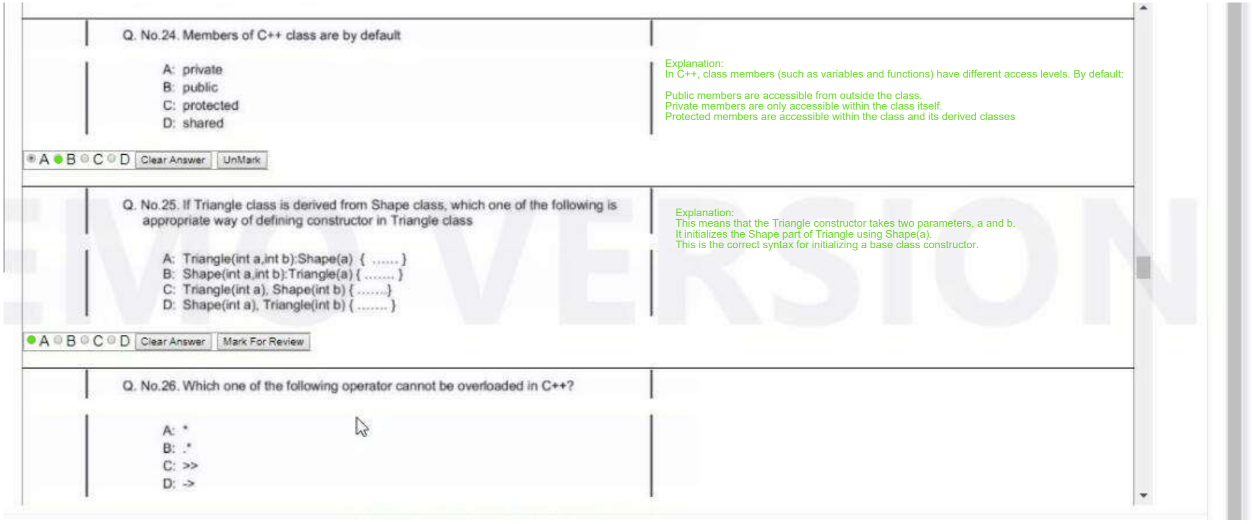
A: Constants
B: Identifiers
C: Reserved words
D: Literals

○ A ○ B ● C ○ D Clear Answer Mark For Review

Explanation:
reserved words (also known as keywords) have predefined meanings and cannot be used
for other purposes, such as naming variables or functions. They form the basic building
blocks of a program's syntax.

Q. No.24. Members of C++ class are by default

A: private
B: public
C: protected
D: shared



Q. No.24. Members of C++ class are by default

A: private
B: public
C: protected
D: shared

○ A ● B ○ C ○ D Clear Answer UnMark

Explanation:
In C++, class members (such as variables and functions) have different access levels. By default:

Public members are accessible from outside the class.
Private members are only accessible within the class itself.
Protected members are accessible within the class and its derived classes

Q. No.25. If Triangle class is derived from Shape class, which one of the following is appropriate way of defining constructor in Triangle class

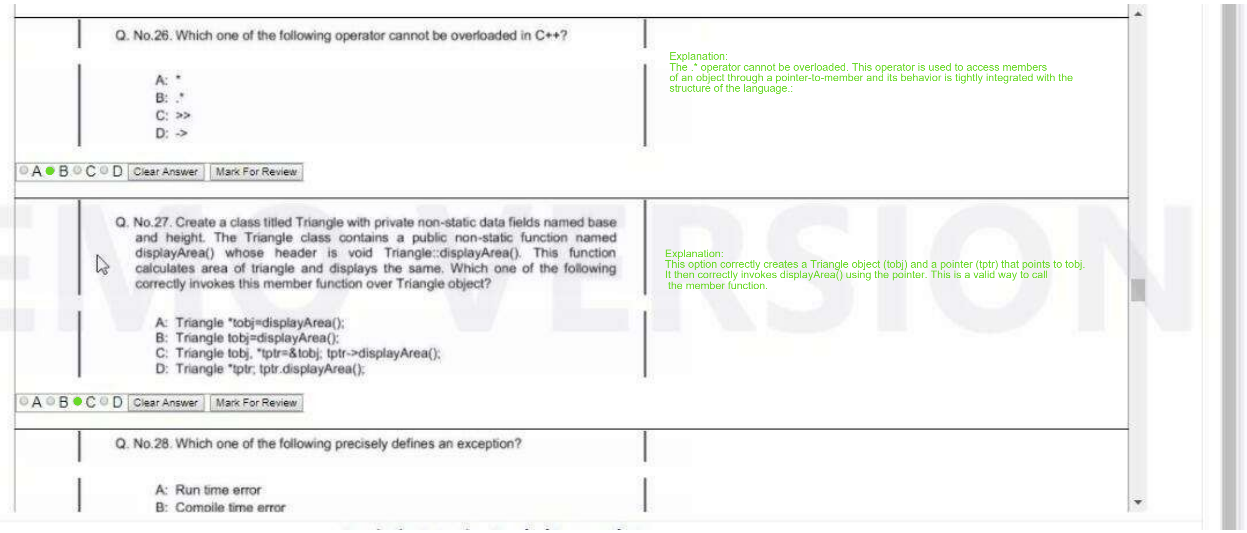
A: Triangle(int a,int b):Shape(a) { ...... }
B: Shape(int a,int b):Triangle(a) { .... }
C: Triangle(int a), Shape(int b) { ....}
D: Shape(int a), Triangle(int b) { .... }

● A ○ B ○ C ○ D Clear Answer Mark For Review

Explanation:
This means that the Triangle constructor takes two parameters, a and b.
It initializes the Shape part of Triangle using Shape(a).
This is the correct syntax for initializing a base class constructor.

Q. No.26. Which one of the following operator cannot be overloaded in C++?

A: *
B: .*
C: >>
D: ->



Q. No.26. Which one of the following operator cannot be overloaded in C++?

A: *
B: .*
C: >>
D: ->

○ A ● B ○ C ○ D

Explanation:
The .* operator cannot be overloaded. This operator is used to access members of an object through a pointer-to-member and its behavior is tightly integrated with the structure of the language.:

Q. No.27. Create a class titled Triangle with private non-static data fields named base and height. The Triangle class contains a public non-static function named displayArea() whose header is void Triangle::displayArea(). This function calculates area of triangle and displays the same. Which one of the following correctly invokes this member function over Triangle object?

A: Triangle *tobj=displayArea();
B: Triangle tobj=displayArea();
C: Triangle tobj, *tptr=&tobj; tptr->displayArea();
D: Triangle *tptr; tptr.displayArea();

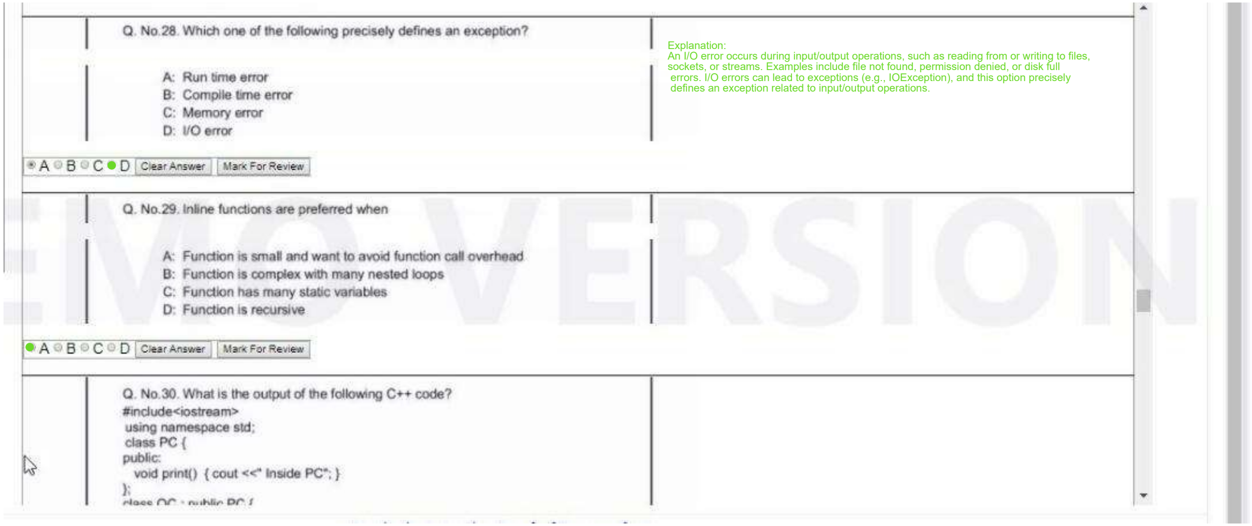
○ A ○ B ● C ○ D

Explanation:
This option correctly creates a Triangle object (tobj) and a pointer (tptr) that points to tobj. It then correctly invokes displayArea() using the pointer. This is a valid way to call the member function.

Q. No.28. Which one of the following precisely defines an exception?

A: Run time error
B: Compile time error

EMO VERSION



Q. No.28. Which one of the following precisely defines an exception?

A: Run time error
B: Compile time error
C: Memory error
D: I/O error

◉ A ○ B ○ C ● D

Explanation:
An I/O error occurs during input/output operations, such as reading from or writing to files, sockets, or streams. Examples include file not found, permission denied, or disk full errors. I/O errors can lead to exceptions (e.g., IOException), and this option precisely defines an exception related to input/output operations.

Q. No.29. Inline functions are preferred when

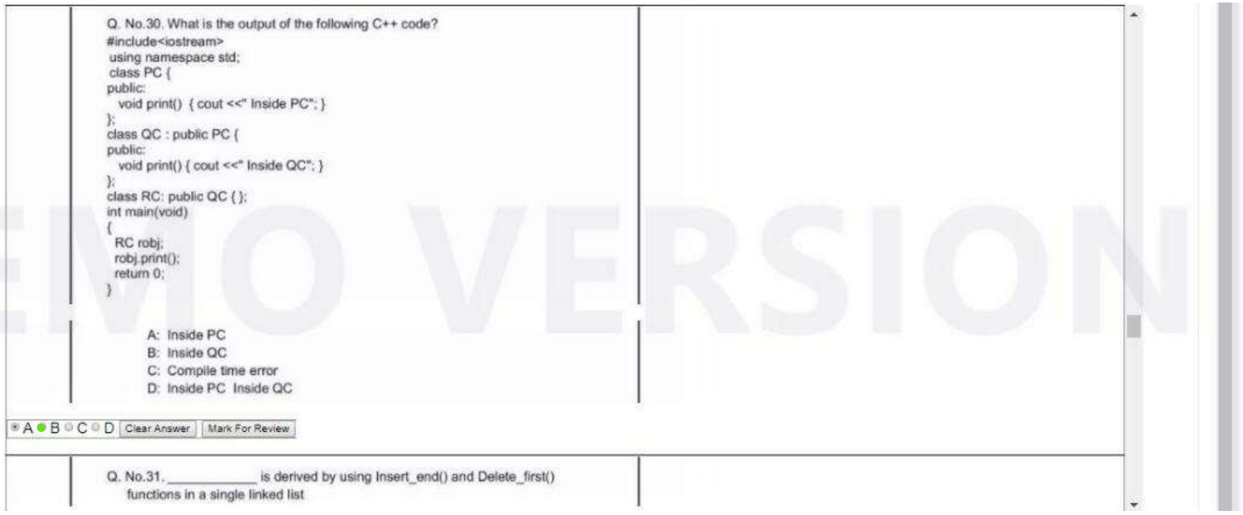
A: Function is small and want to avoid function call overhead
B: Function is complex with many nested loops
C: Function has many static variables
D: Function is recursive

● A ○ B ○ C ○ D

Q. No.30. What is the output of the following C++ code?

```
#include<iostream>
 using namespace std;
 class PC {
public:
   void print()  { cout <<" Inside PC"; }
};
```

EMO VERSION



Q. No.30. What is the output of the following C++ code?

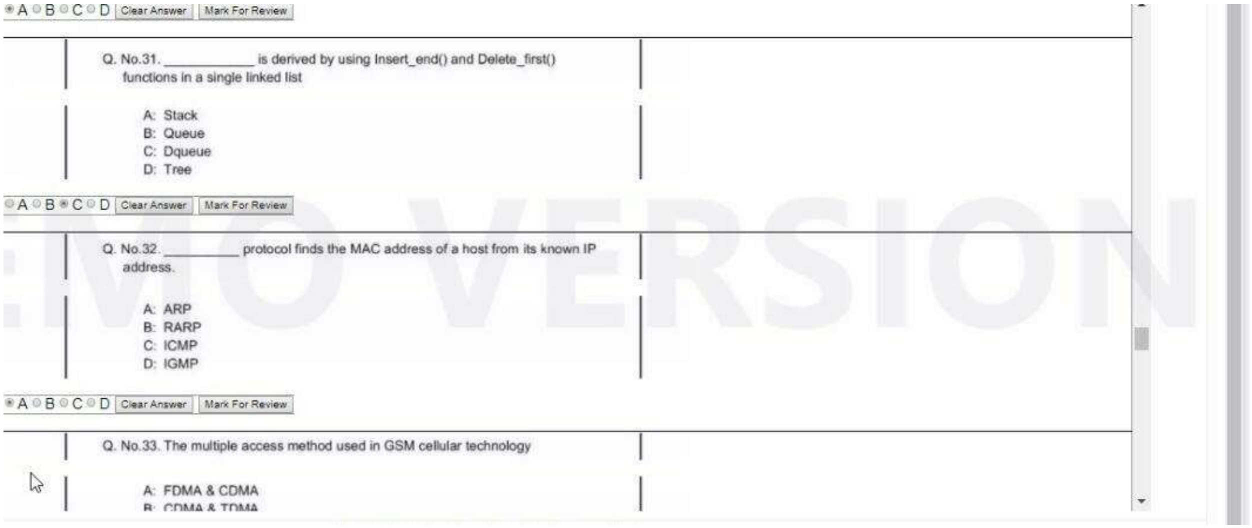
```
#include<iostream>
 using namespace std;
 class PC {
public:
   void print()  { cout <<" Inside PC"; }
};
class QC : public PC {
public:
   void print() { cout <<" Inside QC"; }
};
class RC: public QC { };
int main(void)
{
  RC robj;
  robj.print();
  return 0;
}
```

A: Inside PC
B: Inside QC
C: Compile time error
D: Inside PC Inside QC

◉ A ● B ○ C ○ D

Q. No.31. ____________ is derived by using Insert_end() and Delete_first() functions in a single linked list

EMO VERSION



● A ○ B ○ C ○ D

Clear Answer
Mark For Review

Q. No.31. ____________ is derived by using Insert_end() and Delete_first() functions in a single linked list

A: Stack
B: Queue
C: Dqueue
D: Tree

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.32. __________ protocol finds the MAC address of a host from its known IP address.

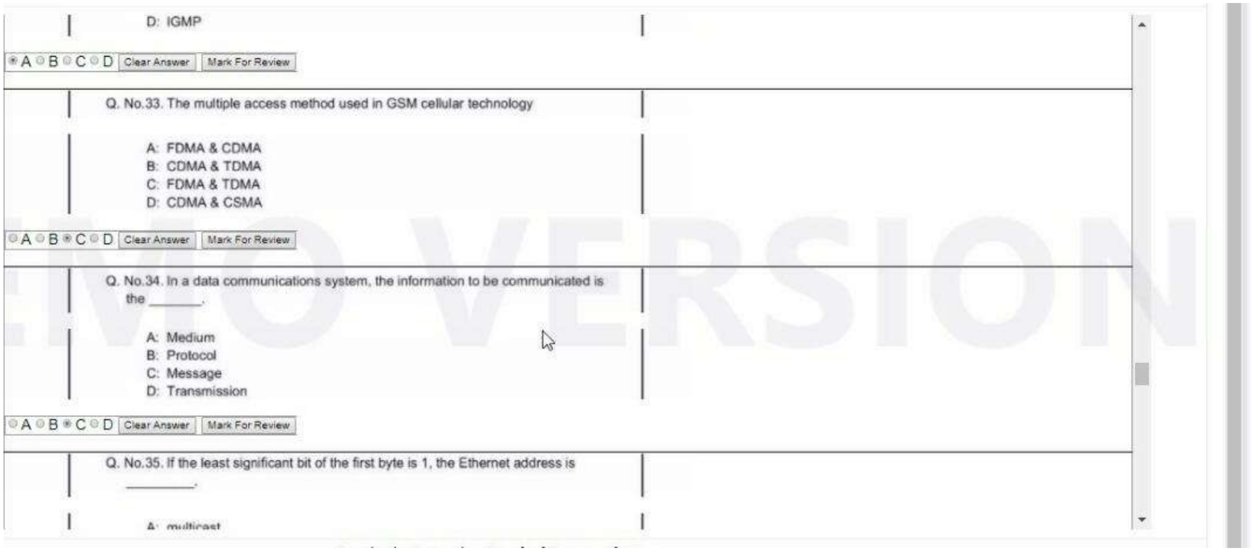
A: ARP
B: RARP
C: ICMP
D: IGMP

● A ○ B ○ C ○ D

Clear Answer
Mark For Review

Q. No.33. The multiple access method used in GSM cellular technology

A: FDMA & CDMA
B: CDMA & TDMA



D: IGMP

● A ○ B ○ C ○ D

Clear Answer
Mark For Review

Q. No.33. The multiple access method used in GSM cellular technology

A: FDMA & CDMA
B: CDMA & TDMA
C: FDMA & TDMA
D: CDMA & CSMA

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.34. In a data communications system, the information to be communicated is the _______.

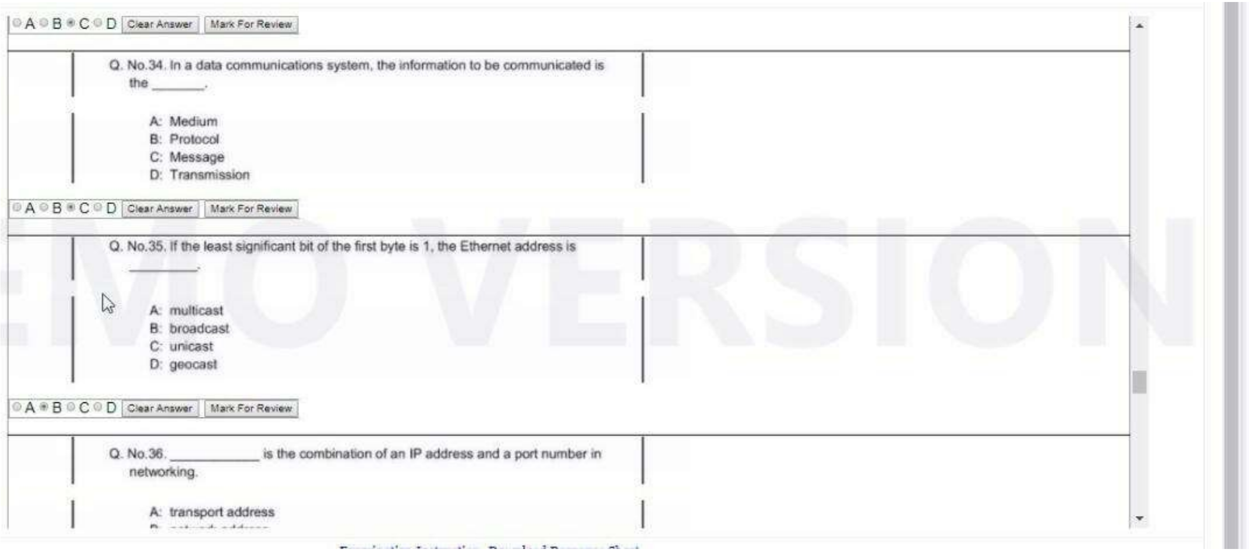
A: Medium
B: Protocol
C: Message
D: Transmission

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.35. If the least significant bit of the first byte is 1, the Ethernet address is _________.

A: multicast



○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.34. In a data communications system, the information to be communicated is the _______.

A: Medium
B: Protocol
C: Message
D: Transmission

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.35. If the least significant bit of the first byte is 1, the Ethernet address is _________.

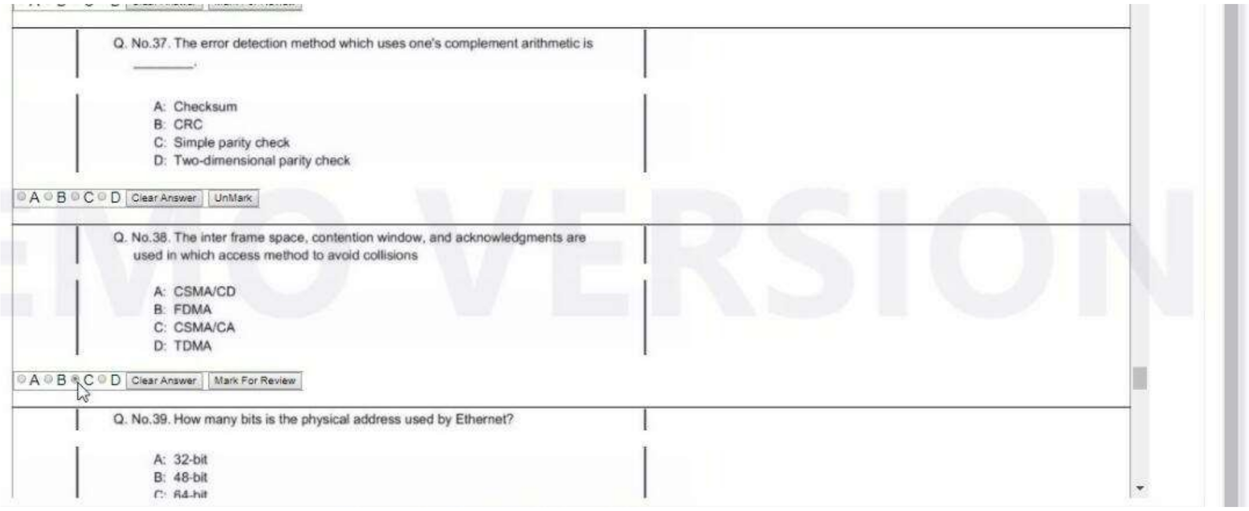
A: multicast
B: broadcast
C: unicast
D: geocast

○ A ● B ○ C ○ D

Clear Answer
Mark For Review

Q. No.36. ___________ is the combination of an IP address and a port number in networking.

A: transport address



Q. No.37. The error detection method which uses one's complement arithmetic is ________.

A: Checksum
B: CRC
C: Simple parity check
D: Two-dimensional parity check

○ A ○ B ○ C ○ D Clear Answer UnMark

Q. No.38. The inter frame space, contention window, and acknowledgments are used in which access method to avoid collisions

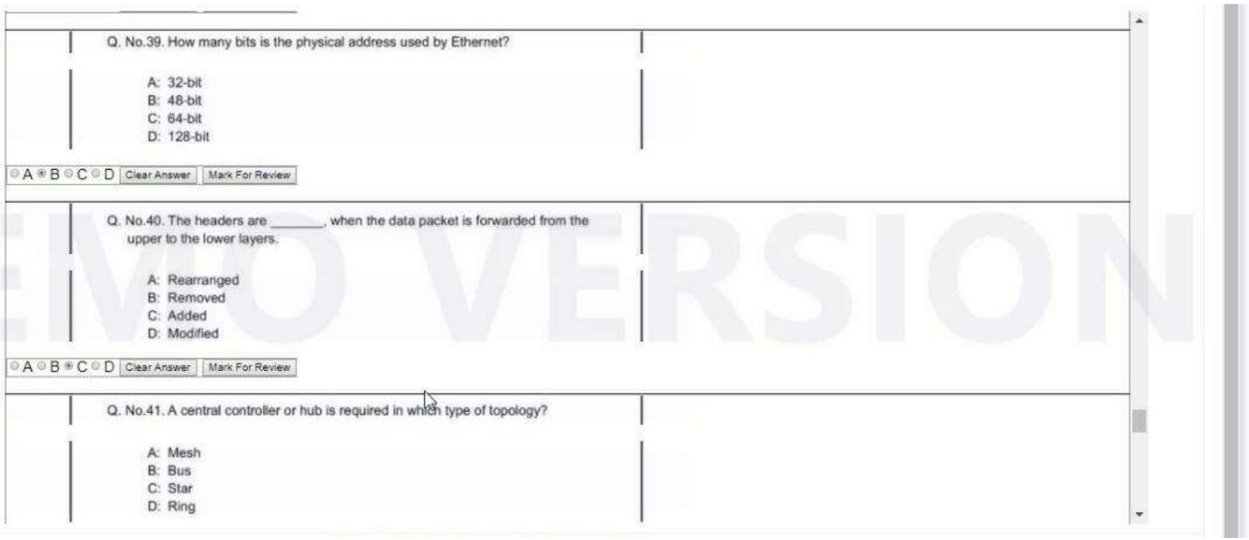
A: CSMA/CD
B: FDMA
C: CSMA/CA
D: TDMA

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.39. How many bits is the physical address used by Ethernet?

A: 32-bit
B: 48-bit
C: 64-bit

EMO VERSION



Q. No.39. How many bits is the physical address used by Ethernet?

A: 32-bit
B: 48-bit
C: 64-bit
D: 128-bit

○ A ● B ○ C ○ D Clear Answer Mark For Review

Q. No.40. The headers are ________, when the data packet is forwarded from the upper to the lower layers.

A: Rearranged
B: Removed
C: Added
D: Modified

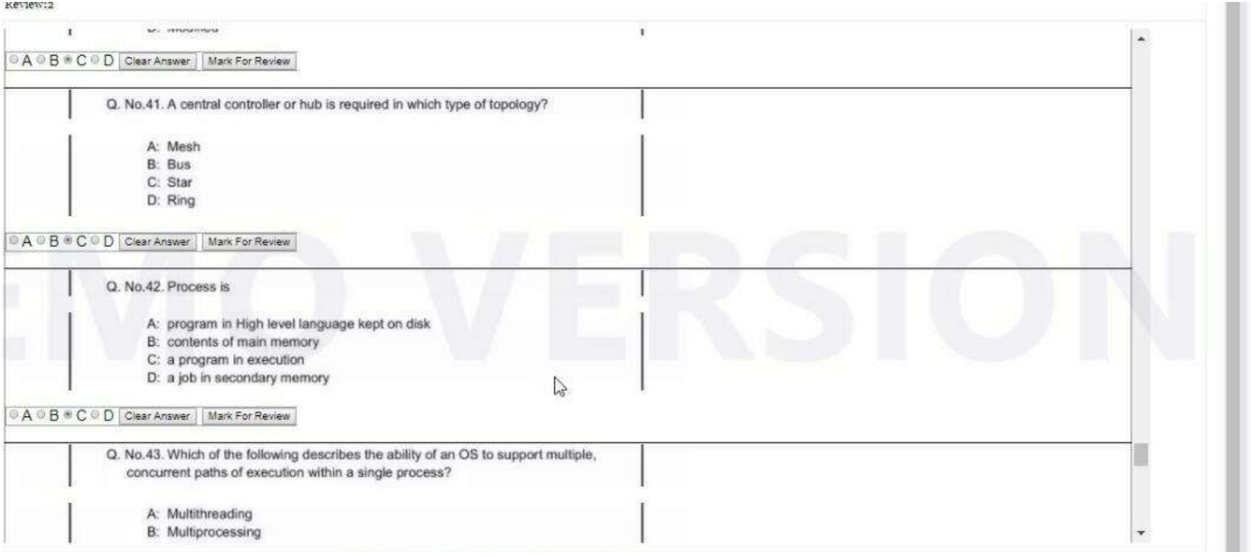
○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.41. A central controller or hub is required in which type of topology?

A: Mesh
B: Bus
C: Star
D: Ring

EMO VERSION

Review:2



○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.41. A central controller or hub is required in which type of topology?

A: Mesh
B: Bus
C: Star
D: Ring

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.42. Process is

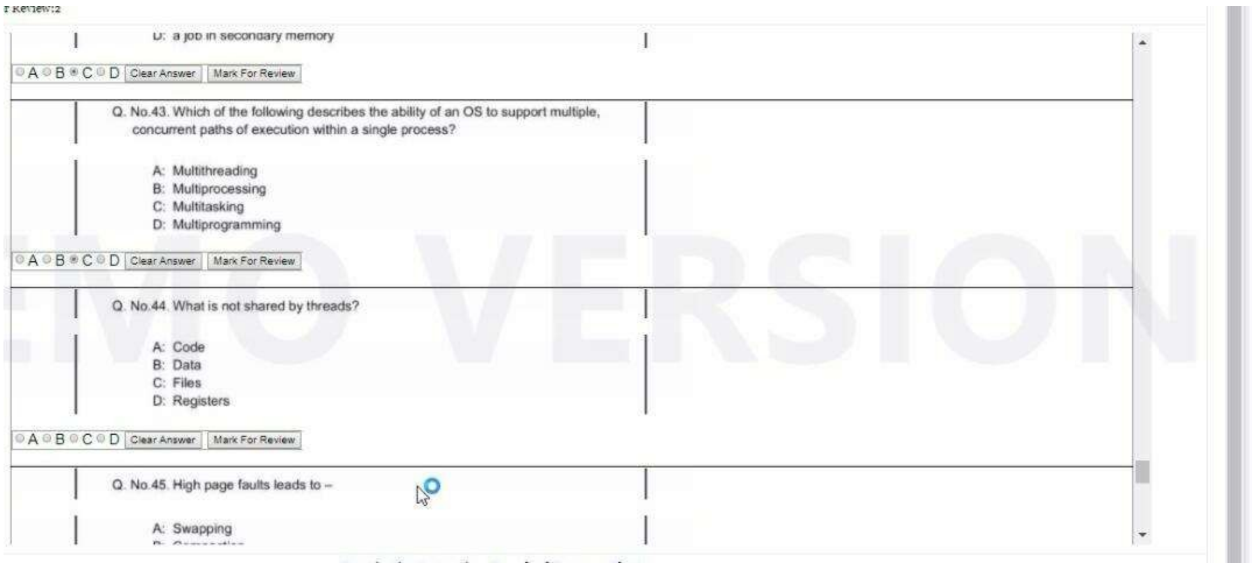
A: program in High level language kept on disk
B: contents of main memory
C: a program in execution
D: a job in secondary memory

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.43. Which of the following describes the ability of an OS to support multiple, concurrent paths of execution within a single process?

A: Multithreading
B: Multiprocessing

EMO VERSION



r Review:2

D: a job in secondary memory

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.43. Which of the following describes the ability of an OS to support multiple, concurrent paths of execution within a single process?

A: Multithreading
B: Multiprocessing
C: Multitasking
D: Multiprogramming

○ A ○ B ● C ○ D Clear Answer Mark For Review

EMO VERSION

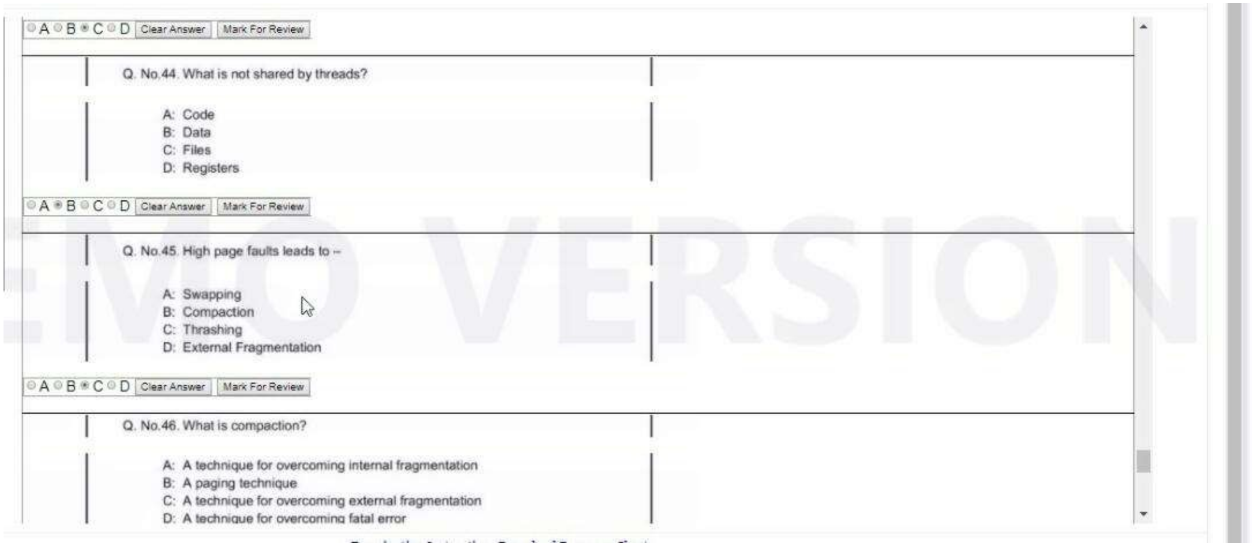
Q. No.44. What is not shared by threads?

A: Code
B: Data
C: Files
D: Registers

○ A ○ B ○ C ○ D Clear Answer Mark For Review

Q. No.45. High page faults leads to –

A: Swapping



○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.44. What is not shared by threads?

A: Code
B: Data
C: Files
D: Registers

○ A ● B ○ C ○ D Clear Answer Mark For Review

EMO VERSION

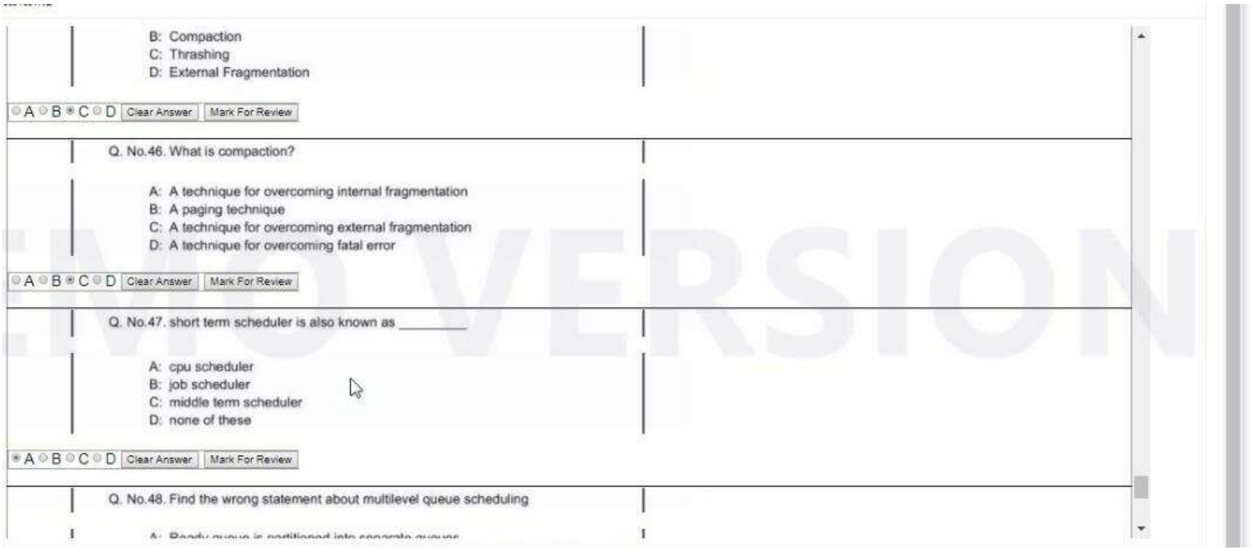
Q. No.45. High page faults leads to –

A: Swapping
B: Compaction
C: Thrashing
D: External Fragmentation

○ A ○ B ● C ○ D Clear Answer Mark For Review

Q. No.46. What is compaction?

A: A technique for overcoming internal fragmentation
B: A paging technique
C: A technique for overcoming external fragmentation
D: A technique for overcoming fatal error



B: Compaction
C: Thrashing
D: External Fragmentation

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

Q. No.46. What is compaction?

A: A technique for overcoming internal fragmentation
B: A paging technique
C: A technique for overcoming external fragmentation
D: A technique for overcoming fatal error

○ A ○ B ● C ○ D

Clear Answer
Mark For Review

EMO VERSION

Q. No.47. short term scheduler is also known as ________

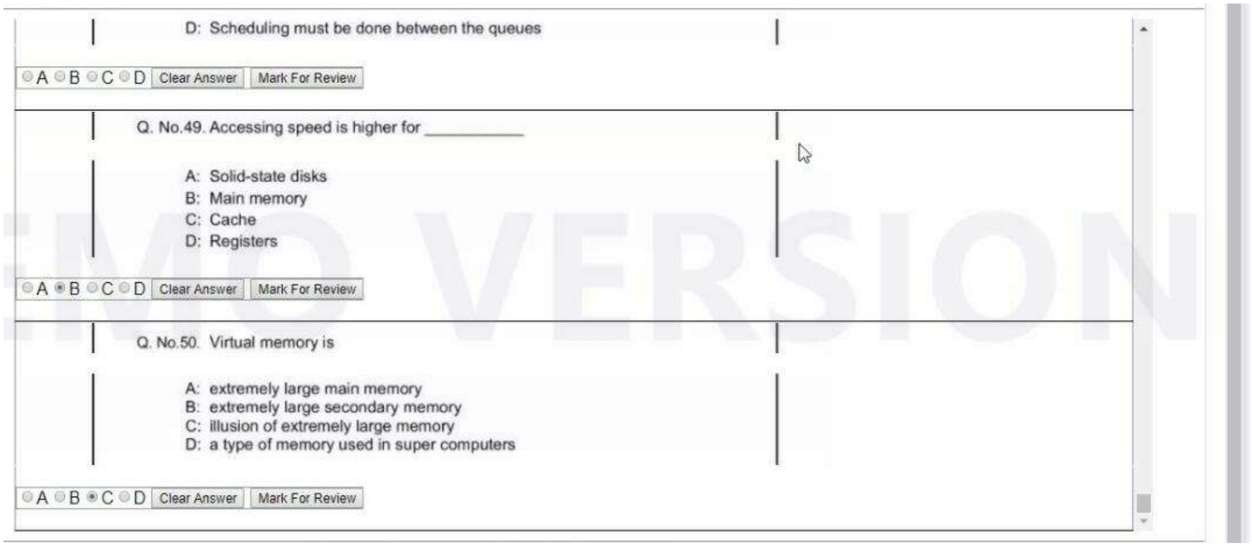
A: cpu scheduler
B: job scheduler
C: middle term scheduler
D: none of these

● A ○ B ○ C ○ D

Clear Answer
Mark For Review

Q. No.48. Find the wrong statement about multilevel queue scheduling

A: Ready queue is partitioned into separate queues



D: Scheduling must be done between the queues

○ A ○ B ○ C ○ D

Clear Answer
Mark For Review

Q. No.49. Accessing speed is higher for ___________

A: Solid-state disks
B: Main memory
C: Cache
D: Registers

○ A ● B ○ C ○ D

Clear Answer
Mark For Review

EMO VERSION

Q. No.50. Virtual memory is

A: extremely large main memory
B: extremely large secondary memory
C: illusion of extremely large memory
D: a type of memory used in super computers

○ A ○ B ● C ○ D

Clear Answer
Mark For Review