Q.11 Assume the following values are inserted into a BST in the given order.

4,5,7,1,2,3,6,9,8,10.

Consider the following function:

```
void tree (Node * root)
{     if (root == NULL) return;
      printf("%d", root → data);
      tree (root → left);
      tree (root → right);
      return;
}
struct node
{     int data;
      struct node * left;
      struct node * right;
} Node;
```

Find the output printed by the above function, if the root pointer of the BST is passed to the "tree" function.

(a) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
(b) 4, 1, 2, 3, 5, 7, 6, 9, 8, 10
(c) 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
(d) 4, 5, 7, 1, 2, 3, 6, 9, 8, 10

Q.12 Consider the following code.

```
int a = 32, b = 2, c = 3;
Switch (X)
{
    Case 2: printf("%d", a);
    Case 4: printf("%d", b);
    Case 6: break;
    Case 8: printf("%d", c);
    default: printf("%d", b);
}
```

Find the missing statement X, if the above 'C' code prints the output as 32.

(a) $b * c$ (b) $b * c - 2$
(c) $b + c * 2$ (d) None of these

Q.13 Which of the following statement is false about 'return' statement?

(a) It terminates the execution of a function.
(b) Control moves back to the calling environment after the return statement execution.
(c) It cannot contain an expression.
(d) It may appear more than once in the same function.

Q.14 Consider the following pseudocode.

```
int i = 0;
main( )
{
    i = 3;
    A( );
    B( );
}
A( ) { print "i"; }
B( ) { int i = 2; A( )}
```

What is the output of the above code if it uses lexical scoping?

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

Q.15 Which of the following is a valid switch statement?

(a)
```
switch (i)        // i is an integer
{
    case 1: break;
    case j: break; // j is a variable
}
```

(b)
```
switch (i)        // i is a string
{
    case "abc" : break;
    case "xyz" : break;
}
```

(c)
```
switch (i)        // i is an integer
{
    case 1 : break;
    case 2 * 4 : break;
}
```

(d) Both (a) and (c)

Q.16 Consider the following code

```
int main( )
{
    char A[.] = "gate";
    int x;
    for (x = 0; A[x]; x++)
    {
        printf("%c", A[x]);
    }
}
```

What is the output printed by the code?

(a) gate (b) g
(c) runtime error (d) compile time error

Q.17 Consider the following code.

```
int f(int a, int b)
{   if (b == 0) return 1;
    else if (b % 2 == 0)
    {
        return (f(a, b/2) * f(a, b/2));
    }
    else { return (a * f(a, b/2) * f(a, b/2)); }
}
```

What is the return value of f(2, 10)?

Q.18 Consider the following *C* program

```
# include <stdio.h>
void f(int x, int * p)
{
     *p = x;
      x = 10;
}
int main ( )
{
     int a = 5,  b = 6;
     int  *p = &a, **q;
     *p = 20; q = &p;
     f(a, &b);
     *q = &b;
     *p = 30;
     printf("%d, %d", a, b);
}
```

What is the output product by above C program

(a) 10, 20 (b) 20, 30

(c) 30, 10 (d) 20, 20

Q.19 What will be the output printed by the following *C* program

```
void main ( )
{
     int x = 1,  i,  y = 2;
     for (i = 0; i < 5;  i++)
     {
           x << 1;
           y = x + i;
     }
     printf ("%d, %d", x, y);
}
```

(a) 1, 5 (b) 32, 5

(c) 1, 72 (d) 32, 72

Q.20 Which of the following is illegal statement in *C*.

(a) `int (**p) [ ];` (b) `int*(*p) ( );`

(c) `int (*f ( )) [ ];` (d) `int*f( ) [ ];`

Q.21 Consider the following recursive *C* functions

```
int f(int i)
{
     if(x == 0)  return 1;
     return f(x – 1) + g(x – 1);
}
int g(int x)
{
     if (x == 0) return 2;
     return g(x – 1) + g(x – 1);

}
```

What is the value returned by f(g (1))?

Q.22 Which of following declarations represents an array of *N* pointers to functions, returning pointers to functions and returning pointer to character?

(a) `char **((*a[M]( )) ( );`

(b) `char **((*a[M]))( )( );`

(c) `char ***((a[M]( ))) ( );`

(d) `char *(*(*a[M])( )) ( );`

Q.23 What is the output of the following code

```
void main( )
{
     int const*p = 5;
     printf("%d", ++(*p));
}
```

(a) 5 (b) 6

(c) 7 (d) Compiler error

Q.24 Consider the following *rec* function.

```
rec (int x)
{
     static int f;
     if (x==1)
          return (1);
     else
          f+ = x * rec (x – 1);
     return (f);
}
```

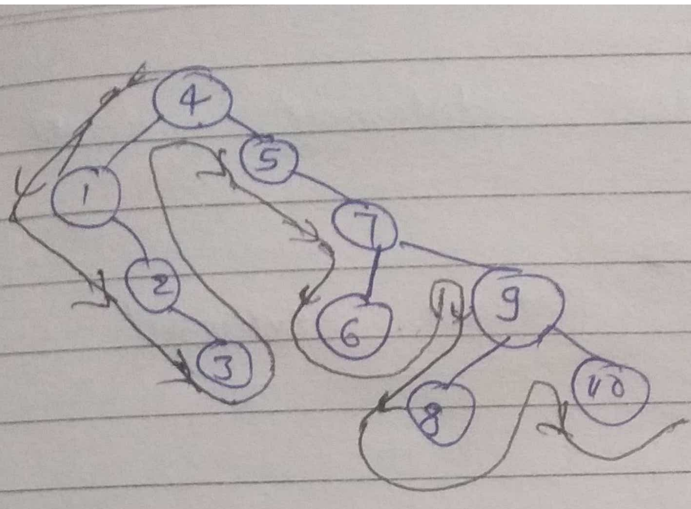
Find the value returned by rec (5).

Q.25 Find the output of the following program.

```
main( )
{    int i =_1_abc (10);
     printf ("%d\ n", – –i);
}
int  _1_abc (int i)
{
     return (i++);
}
```

11→

B



↳ Root , left ; Right

↦ 4, 1, 2, 3, 5, 7, 6, 9, 8, 10

Ans

12 ⇒

C

```
int a=32, b= 2, c=3;
switch (x)
{ case 2:   printf("%d", a);
    case 4:   printf("%d", b);
    case 6:      break
    case 8:   printf("%d", c);  }  → 32
      default: printf("%d", b);
}
```

x=3 ⇒ o/p = 32

x=8 →

~~A~~ b*c = 6

~~B~~ b*c − 2 = 4

C b*c*2 = 8

13 ⇒ False statement - about return

C

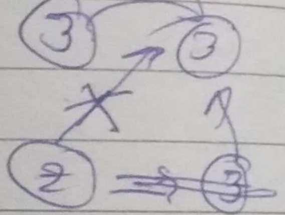
Return statement can not an expression

Ans

14 ⇒

D

```
int i=0;
main()
{ i=3;  A();  B(); }
A()
{ print(i); }
B()
{ int i = 2; A() }
```



O/P = 3, 3 Ans

15 ⇒ valid switch =

C

```
switch (i)
{
    Case 1: break;
    cas 2*4: break;
}
```

16⇒

A

```
int main()
{
    char A[] = "gate";
    int x;
    for (x=0; A[x]; x++)
    { printf("%c", A[x]); }
}
```

A[0] = "g" Ans

O/P = gate

17⇒

1024

```
int f(int a, int b)
{
    if(b==0) return 1;
    else if(b%2==0) return (f(a,b/2) * f(a, b/2));
    else return (a* f(a, b/2) * f(a, b/2));
}
```

$f(2, 10) = 2^{10} = 1024$ Ans

18⇒

B

```
#include<stdio.h>
void f( int x, int *p)
{ *p = x; x = 10; }
int main()
{ int a = 5, b = 6; int *p = &a, **q;
  *p = 20; q = &p; f(a, &b); *q = &b; *p = 30;
  printf("%d %d", a, b);
}
```

20, 30 Ans

(19) ⇒ (A)

```
void main()
{
    int x=1, i, y=2;
    for(i=0; i<5; i++)
    { x<<1; y=x+i; }
    printf("%d %d", x, y);
}
```

not change value of (x)

0 =
1 = , 1
2 = 64 , 2
3 = 8 , 4 , 2
4 = 16 , 7
32 ,

$O/P = x=1, y=5$ Ans

(20) ⇒ (D)

illegal statement in C

`int *f()[];` → not valid

(21) ⇒ (31)

```
int f(int i)
{
    if(x==0) return 1;
    return f(x-1) + g(x-1);
}
int g(int x)
{ if (x==0) return 2;
    return g(x-1) + g(x-1);
}
```

→ f(g(1)) = 31

$f(4) = f(3)+g(3)$

$= f(2)+g(2)+g(3)$

$= f(1)+g(1)+g(2)+g(3)$

~~= 1 + 2 + 2×2 + 2×2~~

~~1+4+8+16~~

$f(0)+g(0)+g(1)+g(2)+g(3)$

$= 1+2+4+8+16$

$= 31$ Ans

22⇒ array of N pointers to f^n's, returning pointers to f^n's returning pointer to character—

D

→ char *

23⇒

D

```
void main()
{
    int const *p = 5;
    printf("%d", ++(*p));
}
```

→ Compile Error

24⇒

240

```
rec(int x)
{ static int f;
    if (x == 1)
        return 1;
    else
        f += x * rec(x-1);
    return(f);
}
```

→ rec(5) = f+5 * rec(4) = f+4*(rec(3) → ←

40 8

= 40 + 5×40

= 240 Ans

25

9

```
main()
{   int  i= _1_abc (10);
       printf("%d\n", --i);
 }
 int    _1_abc (int i)
 {
        return  i++;
  }
```

O/P = 9 Ans