



c. Sum of a, b

d. minimum of a, b

7. If an integer needs two bytes of storage then maximum value of an unsigned integer is

a.  $2^{16} - 1$

b.  $2^{15} - 1$

c.  $2^{16}$

d.  $2^{15}$

8. `Printf ( "%d", printf ("tim"));`

a. Results in a syntax error

b. Outputs tim3

c. Outputs garbage

d. Prints tim and terminates abruptly

9. A function q that accepts a pointer to a character as argument and returns a pointer to an array of integer can be declared as

a. `int ( * q ( char * ) ) []`

b. `int * q ( char * ) []`

c. `int (*q) ( char *)[]`

d. None of these

10. `a << 1` is equivalent to

a. Multiplying a by 2

b. Dividing a by 2

c. Adding 2 to a

d. None of the above

11. In a certain machine, the sum of an integer and its 1's complement is  $2^{20} - 1$ . Then `sizeof(int)`, in bits, will be

a. 16

b. 32

c. Unpredictable

d. None of the above

12. If abc is the input then the following program fragment results in

```
char x, y, z;
```

```
printf ("%d", scanf ("%c%c%c", &x, &y, &z));
```

 results in

a. A syntax error

b. A fatal error

c. Segmentation violation

d. Printing of 3

13. Consider the statements:

```
putchar ( getchar());
```

```
putchar (getchar());
```

If

a

b

Is the input, the output will be

a. An error message

b. This can't be the input

- c. ab
- d. a b

14. let a, b be two positive integers. Which of the following options correctly relates / and % ?

- a.  $b = (a/b) * b + a\%b$
- b.  $a = (a/b) * b + a\%b$
- c.  $b = (a\%b) * b + a/b$
- d.  $a = (a\%b) * b + a/b$

15. if the word size is 16 bit then  $\sim 0xc5$  will be

- a. 0x3a
- b. 0xff3a
- c. 0x5c
- d. None of the above

16. Which of the following operations produce an 1, if the input bits are 1 and 1?

- a. OR
- b. And
- c. Exclusive or
- d. Exclusive nor

17. The number of possible values of m, such that  $m \& 0x3f$  equals 0x23 is

- a. 1
- b. 2
- c. 3
- d. 4

18. `calloc(m,n)`; is equivalent to

- a. `malloc(m*n, 0)`;
- b. `memset(0, m*n)`;
- c. `ptr = malloc (m*n); memset (p, 0, m*n)`;
- d. `ptr = malloc (m*n); strcpy(p,0)`;

19. consider the following program fragment:

```
char c = 'a' ;  
while (c++ <= 'z')  
    putchar (xxx)
```

if the required output is *abcdefghijklmnopqrstuvwxy* then xxx should be

- a. c
- b. c++
- c. c-1
- d. -c

20. The following code fragment

```
int x, y = 2, z, a;  
x = (y *= 2) + ( z = a = y ) ;  
printf ( "%d", x);
```

- a. Prints 8
- b. Prints 6
- c. Prints 6 or 8 depending on the compiler implementation

- d. Is syntactically wrong
21. If  $n$  has value 3 then the output of the statement `printf( "%d%d", n++, ++n);`
- Is 3 5
  - Is 4 5
  - Is 4 4
  - Is implementation dependent
22. If a variable can take only integral values from 0 to  $n$ , where  $n$  is a constant integer, then the variable can be represented as a bit – field whose width is the integral part of (the log in the answers are to the base 2)
- $\log(n) + 1$
  - $\log(n - 1) + 1$
  - $\log(n + 1) + 1$
  - None of the above
23. The statement `printf("%d", 10?0?5:11:12);` prints
- 10
  - 0
  - 12
  - 11
24. The statement `printf( "%d", 9a++);` prints
- The current value of  $a$
  - The current value of  $a+1$
  - An error message
  - Garbage
25. The statement `printf("%d", ++5);` prints
- 5
  - 6
  - An error message
  - Garbage
26. If  $p$  is a pointer to an integer and  $t$  is a pointer to a character then `sizeof(p)` will be
- Same as that of `sizeof(t)`
  - Greater than that of `sizeof(t)`
  - Less than that of `sizeof(t)`
  - None of the above
27. Consider the declaration:- `char street [10] = "abcdefghi";` choose the correct remarks(s).
- `&street` and `street` will have different values
  - `&street` is meaningless
  - `&street+1` and `street + 1` will have the same values
  - None of the above
28. `x -= y + 1;` does the same as
- `x = x - y + 1`
  - `x = - x - y - 1`
  - `x = - x + y + 1`
  - `x = x - y - 1`
29. `printf( "%c", 100);`

- a. Prints 100
- b. Prints ASCII equivalent of 100
- c. Prints garbage
- d. None of the above

30. The following statement

```
printf( "%f", 9/5);
```

 prints

- a. 1.8
- b. 1.0
- c. 2.0
- d. None of the above

31. The following program fragment

```
for (i = 3 ; i < 15; i += 3) ;  
printf( "%d", i);
```

results in

- a. A syntax error
- b. An execution error
- c. Printing of 12
- d. Printing of 15

32. The following program fragment

```
for ( i = 1; i < 5; ++i)  
if ( i == 3) continue;  
else printf ( "%d", i);
```

 results in the printing of

- a. 1 2 4 5
- b. 1 2 4
- c. 2 4 5
- d. None of the above

33. The following program fragment

```
if ( a = 0 )  
printf ( "a is zero" );  
else  
printf ( " a is not zero ");
```

results in the printing of

- a. a is zero
- b. a is not zero
- c. nothing
- d. garbage

34. The following program fragment

```
if ( a = 7 )  
printf ( "a is seven" );  
else  
printf ( " a is not seven");
```

results in the printing of

- a. a is seven
- b. a is not seven
- c. nothing

d. garbage

35. The following loop

```
for ( i = 1, j = 10; i < 6; ++ i, -- j )  
    printf ( "%d %d", i, j );
```

prints

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

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

c. 1 1 1 1 1 9 9 9 9 9

d. None of the above

36. The following program fragment

```
int a = 4, b = 6;  
printf ( "%d", a == b );
```

a. Outputs an error message

b. Prints 0

c. Prints 1

d. None of the above

37. The following program

```
main()  
{  
    int i = 5;  
    if ( i == 5 ) return;  
    else printf ( " i is not five ");  
    printf ( "over");  
}
```

Results in

a. A syntax error

b. An execution error

c. Printing of over

d. Execution termination, without printing anything

38. The following program fragment

```
int i = 107, x = 5;  
printf ( ( x > 7 ) ? "%d" : "%c", i);
```

results in

a. An execution error

b. A syntax error

c. Printing of k

d. None of the above

39. The following loop

```
while ( printf ( "%d", printf ( "az")))  
    printf ( "by" );
```

a. Prints azbybybyby....

b. Prints azbyazbyazbyazby.....

c. Results in a syntax error.

d. None of the above

40. The following statements

```
for ( i = 3; i < 15; i += 3)
{
    printf( "%d", i );
    ++i;
}
```

Will result in the printing of

- a. 3 6 9 12
- b. 3 6 9 12 15
- c. 3 7 11
- d. 3 7 11 15

41. If  $a = 9$ ,  $b = 5$  and  $c = 3$ , then the expression  $( a - a/b * b \% c ) > a\%b\%c$  evaluates to

- a. True
- b. False
- c. Invalid
- d. 0

42. Consider the following program fragment

```
if ( a > b )
if ( b > c )
s1;
else s2;
```

s2 will be executed if

- a.  $a \leq b$
- b.  $b > c$
- c.  $b \leq c$  and  $a \leq b$
- d.  $a > b$  and  $b \leq c$

43. The following program fragment

```
if ( 2 < 1 )
;
else
x = ( 2 < 0 ) ? printf ( "one" ) : printf ( "four" );
printf ( "%d", x);
```

- a. Prints nothing
- b. Results in a syntax error
- c. Prints four0
- d. None of the above

44. The following program fragment

```
int x = 4, y = x, i;
for ( i = 1 ; i < 4; ++ i)
    x += x;
```

Outputs an integer that is same as

- a.  $8 * y$

b.  $y * (1 + 2 + 3 + 4)$

c.  $y * 4$

d.  $y * y$

45. Consider the declaration

```
static char hello[] = "hello";
```

The output of `printf("%s\n", hello);`

Will be the same as that of

a. `puts("hello");`

b. `puts(hello);`

c. `printf("%s\n", "hello");`

d. `puts("hello\n");`

46. The following program fragment

```
int x[5][5], i, j;
```

```
for ( i = 0; i < 5; j++)
```

```
    for( j = 0; j < 5 ; j++)
```

```
        x[i][j] = x[j][i];
```

a. Transposes the given matrix *x*

b. Makes the given matrix *x*, symmetric

c. Doesn't alter the matrix *x*

d. None of the above

47. Consider the following type definition

```
typedef char x[10];
```

```
x myArray[5];
```

what will `sizeof(myArray)` be? (Assume one character occupies 1 byte)

a. 15 bytes

b. 10 bytes

c. 50 bytes

d. 30 bytes

48. The following program

```
main()
```

```
{
```

```
    Static int a[] = {7, 8, 9};
```

```
    Printf ( "%d", 2[a] + a[2]);
```

```
}
```

a. Results in bus error

b. Results in segmentation violation error

c. Will not compile successfully

d. None of the above

49. The following program fragment

```
int m, n, b = m = n = 8;
```

```
char wer [80];
```

```
sprint(wer, "%d%d%d", m, n, b);
```

```
puts(wer);
```

a. Prints the string 8 8 8



- b. Prints the null string
- c. Prints the string 888
- d. None of the above

50. The following program

```
Main()
{
    Static char a[3][4] = {"abcd", "mnop", "fghi"};
    Puchar( **a);
}
```

- a. Will not compile successfully
- b. Results in run-time error
- c. Prints garbage
- d. None of the above

51. The default parameter passing mechanism is

- a. Call by value
- b. Call by reference
- c. Call by value result
- d. None of the above

52. If n has value 3, then the statement a[++n] = n++;

- a. Assigns 3 to a[5]
- b. Assigns 4 to a[5]
- c. Assigns 4 to a[4]
- d. Assigns a compiler-dependent value to a[5]

53. The following program

```
main()
{
    int i = 2;
    {
        int i = 4, j = 5;
        printf("%d%d", i, j);
    }
    printf("%d%d", i, j);
}
```

- a. Will not compile successfully
- b. Prints 4525
- c. Prints 2525
- d. None of the above

54. The following program

```
main()
{
    inc(); inc(); inc();
}
inc()
```

```
{
    static int x;
    printf("%d", ++x);
}
```

- a. Prints 012
  - b. Prints 123
  - c. Prints 3 consecutive, but unpredictable numbers
  - d. Prints 111
55. `printf("ab", "cd", "ef");`  
Prints
- a. ab
  - b. abcdef
  - c. abcdef, followed by garbage
  - d. None of the above
56. Virtual memory is
- a. An extremely large main memory
  - b. An extremely large secondary memory
  - c. An illusion of an extremely large memory
  - d. A type of memory used in super computers
57. Page fault occurs when
- a. The page is corrupted by application software
  - b. The page is in main memory
  - c. The page is not in main memory
  - d. One tries to divide a number by 0
58. In a multi – user operating system, 20 requests are made to use a particular resource per hour, on an average. The probability that no requests are made in 45 minutes is.
- a.  $e^{-15}$
  - b.  $e^{-5}$
  - c.  $1 - e^{-5}$
  - d.  $1 - e^{-10}$
59. The expression  $4 + 6 / 3 * 2 - 2 + 7 \% 3$  evaluates to
- a. 3
  - b. 4
  - c. 6
  - d. 7
60. Who is the father of computer?
- a. William Wellsworth
  - b. Frank Babbage
  - c. Charles Babbage
  - d. Lady Ada Lovelace
61. COMPUTER stands for
- a. Common Multi Purpose Utility Terminal
  - b. Common Operating Machine Purposely Used for Technological and Educational Research
  - c. Central Monitoring Poster

- d. None of the above
62. Which one of the following is the output device?
- a. Mouse
  - b. Joystick
  - c. Scanner
  - d. Monitor
63. Which operating system can be used for weather forecasting?
- a. Batch OS
  - b. Real Time OS
  - c. Multiuser OS
  - d. None of the above
64. Stored Memory Concept for digital computer system is given by?
- a. Charles Babbage
  - b. Alan Turing
  - c. Von Neumann
  - d. None of the above
65. Which of the following memory is the fastest till date?
- a. Cache memory
  - b. RAM
  - c. Registers
  - d. Secondary Memory
66. The component of CPU which handles all the computations is known as?
- a. CU
  - b. ALU
  - c. Memory
  - d. None of the above
67. PCB stands for
- a. Printed Circuit Board
  - b. Printed Circuit Breaker
  - c. Post Common Bios
  - d. None of the above
68. EEPROM Stands for
- a. Electrically Erasable Programmable Read Only Memory
  - b. Electrically Erasable Persistent Read Only Memory
  - c. Enhanced Erasable Programmable Read Only Memory
  - d. Electrically Enable Programmable Read Only Memory
69. The process by which data is written onto a magnetic disk is called
- a. Data Writing Technology
  - b. Enhanced Magnetic Simulation
  - c. Magnetic Entropy
  - d. None of the above
70. CPU is made up of
- a. Metals
  - b. Conductors
  - c. Semi-Conductors

d. None of the above