1. **Which of the following is a valid expression in C?**
   1. int basic_pay = 23000;
   2. int basic_pay = 23,000;
   3. int basic pay = 23000;
   4. int $basic_pay = 23000;

   Show Answer

   Answer: A

   Variable names cannot contain white spaces, commas, or $

2. **Which of the following arithmetic operator takes only integer operands?**
   1. +
   2. –
   3. /
   4. %

   Show Answer

   Answer: D

   The modulo division (%) operator produces the remainder of an integer division.

3. **Which of the following operator has the lowest priority?**
   1. &&
   2. +
   3. *
   4. !==

   Show Answer

   Answer: A

4. **Which among the following operator has the highest priority?**
   1. Comma operator
   2. Conditional operator
   3. Relational operator
   4. Unary plus operator

   Show Answer

   Answer: D

5. **Which of the following statement is true?**
   1. All arithmetic operators have the same level of precedence.
   2. The modulus operator % can be used only with integers.
   3. A unary expression consists of only one operand with no operators.
   4. The operators <=, >=, and != all enjoy the same level of priority.

   Show Answer

   Answer: B

   The modulus operator % can be used only with integers.

6. **What will be the output for the following C code?**
   1. 2
   2. 1
   3. 3
   4. Compile-time error

```c
#include <stdio.h>
int main()
{
    int n = 8;
    n = n / 3;
    printf("%d", n);
    return 0;
}
```

   Show Answer

   Answer: A

   2

7. **What is the correct order of precedence of arithmetic operators from highest to lowest?**
   1. %, *, /, +, −
   2. -, +, *, /, %
   3. %, +, -, *, /
   4. +, − , %, *, /

   Show Answer

   Answer: A

8. What will be the output for the following C code?

1. Value of x is 3
2. Value of x is 2
3. Value of x is 1
4. Compile-time error

```c
#include <stdio.h>
void main()
{
    int y = 4;
    int x = 9 % 2 * 4 / 2;
    printf("Value of x is %d", x);
}
```

Show Answer

Answer: A

```
Value of x is 2
```

9. Which of the following is not an arithmetic operation?

1. a*=5;
2. a/=5;
3. a!=5;
4. a%=5;

Show Answer

Answer: C

10. What will be the output for the following C code?

1. 10.6
2. 10
3. 11
4. 16

```c
#include <stdio.h>
int main()
{
    int a = 8;
    double b = 2.6;
    int c;
    c = a + b;
    printf("%d", c);
```

```
}
```

11. **What is the result of a logical or relational expression in C?**
    1. 0 or 1
    2. True or false
    3. 0 if false and a positive number if true.
    4. T or F

12. **What will be the output for the following C code?**
    1. 23
    2. 15
    3. Undefined behavior
    4. 21

```c
#include <stdio.h>
int main()
{
    int x = 5, y = 2;
    x *= x - y;
    printf("%d", x);
    return 0;
}
```

13. **What will be the output for the following C code?**
    1. 4
    2. 2
    3. 1
    4. Undefined behavior

```c
#include <stdio.h>
int main()
{
    int x = 4, y = 2;
    x /= x / y;
    printf("%d\n", x);
    return 0;
}
```

Show Answer

Answer: B

```
2
```

14. **What will be the output for the following C code?**
    1. Compile-time error
    2. Yes
    3. No
    4. Nothing

```c
#include <stdio.h>
void main()
{
    int x = 0;
    if (x = 0)
        printf("Yes");
    else
        printf("No");
}
```

Show Answer

Answer: C

```
No
```

15. **What is the use of sizeof() operator?**
    1. To get the size of data types or variables in bytes
    2. To get the size of variables only
    3. To get the size of the structure only
    4. None of the above

Show Answer

Answer: A

16. What is the operator ':?' called?

1. If-Else operator
2. Ternary operator
3. Logical operator
4. Comparison operator

Show Answer

Answer: A

The ':?' operator is called the conditional operator or ternary operator.

17. Which of the following are true regardless of the implementation?

1. `sizeof(int)` is not less than `sizeof(long)`
2. `sizeof(int)` equals `sizeof(unsigned)`
3. `sizeof(double)` is not less than `sizeof(float)`
4. Both B and C

Show Answer

Answer: D

In C,

`sizeof(int)` is not less than `sizeof(long)` and `sizeof(int)` equals `sizeof(unsigned)` regardless of the implementation,

18. If y is of integer type then the expressions `3 * ( y -8 ) / 9` and `(y - 8) /9 * 3`

1. must yield the same value
2. must yield different values
3. may or may not yield the same value
4. none of the above

Show Answer

Answer: C

In C,

`sizeof(int)` is not less than `sizeof(long)` and `sizeof(int)` equals `sizeof(unsigned)` regardless of the implementation,

19. **Integer division results in**
    1. truncation
    2. rounding
    3. overflow
    4. none of the above

Show Answer

Answer: A

For example, integer division of 8 and 3 will result in 2.

20. **In an expression involving || operator, evaluation**
    1. will be stopped if one of its components evaluates to false
    2. will be stopped if one of its components evaluates to true
    3. takes place from left to right
    4. Both B and C

Show Answer

Answer: D

21. **Implicit type conversion takes place**
    1. across an assignment operator.
    2. if an operator has operands of different data types.
    3. Both A and B
    4. None of the above.

Show Answer

Answer: C

22. **Choose the correct statements**
    1. Casting refers to implicit type conversion
    2. Coercion refers to implicit type conversion
    3. Casting refers to explicit type conversion
    4. Both B and C

Show Answer

Answer: D

23. **Pick the operators that associate from the left**
    1. +
    2. ,
    3. ==
    4. All of above

Show Answer

Answer: D

24. **Pick the operators that associate from the right**
    1. ?:
    2. +=
    3. =
    4. All of above

Show Answer

Answer: D

25. **Pick the operators whose meaning is context-dependent.**
    1. *
    2. #
    3. &
    4. All of above

Show Answer

Answer: D

26. `x -= y + 1;` **means**
    1. x = x − y + 1
    2. x = -x − y − 1
    3. x = -x + y + 1
    4. x = x − y − 1

Show Answer

Answer: D

x -= y + 1 is equivalent to x = x − (y + 1). That is x = x − y − 1.

27. **Which of the following comments about the ++ operator are correct?**
    1. It is a unary operator
    2. The operand can come before or after the operator
    3. It cannot be applied to an expression
    4. All of above

> Show Answer
>
> **Answer: D**

28. **The expression** `4 + 6 / 3 * 2 - 2 + 7 % 3` **evaluates to**
    1. 3
    2. 4
    3. 6
    4. 7

> Show Answer
>
> **Answer: D**

29. **If a=5 and b=2 then** `printf("%d", a+++b);` **will?**
    1. results in a syntax error
    2. print 8
    3. print 10
    4. None of the above

> Show Answer
>
> **Answer: B**
>
> The compiler will tokenize a+++b as a, ++, +, b. So, a+++b is equivalent a++ + b, which evaluates to 8.

30. **The statement** `printf("%d", (a++));` **will print**
    1. the current value of a
    2. the value of a + 1
    3. an error message
    4. garbage

> Show Answer
>
> **Answer: A**

31. **The statement** `printf("%d", ++5);` **will print**
    1. 5
    2. 6
    3. an error message
    4. garbage

Show Answer

Answer: C

32. **The statement** `printf("%d", sizeof(""));` **will print**
    1. 0
    2. 1
    3. an error message
    4. garbage value

Show Answer

Answer: B

33. **The ascending order of precedence of the bit-wise operators &, ^, and | is**
    1. & ^ |
    2. ^ & |
    3. | ^ &
    4. & | ^

Show Answer

Answer: A