

1. Read problem statement 1 and Read 2 integer values to perform bitwise operations $\&$, $|$, $<<$, $>>$, \sim , \wedge .

Ans:- The assignment specifies reading two integer values and performing bitwise operations: $\&$ (AND), $|$ (OR), $<<$ (left shift), $>>$ (right shift), \sim (NOT), \wedge (XOR).

The example input values are $a=4$ and $b=2$.

To solve this, a program would need to:

1. Read two integer values, in this case, 4 and 2.
2. Perform each of the specified bitwise operations on these values.
3. Print the results of each operation.

Here is a breakdown of the bitwise operations and their results using the provided input values, $a=4$ and $b=2$.

The binary representation of $a=4$ is 0100 and $b=2$ is 0010.

* Bitwise AND (&):- Results in 1 if both bits are 1. For $4 \& 2$ (0100 & 0010), the result is 0000, or 0.

$4 \& 2$ is 0

* Bitwise OR (|):- Results in 1 if at least one bit is 1. For $4 | 2$ (0100 | 0010), the result is 0110, or 6.

$4 | 2$ is 6

* Bitwise left shift (<<):- Shifts bits to the left, multiplying by a power of 2. $4 << 1$ (0100 << 1) is 1000, or 8. $4 << 2$ is 16.

* Right shift ($>>$): shifts bits to the right, dividing by a power of 2. $4 >> 1$ ($0100 >> 1$) is 0010 , or 2.
 $4 >> 2$ is 1.

* Bitwise NOT (\sim): Inverts all bits. For a 32-bit integer, ~ 4 results in -5.

* Bitwise XOR (\wedge): Results in 1 if the bits are different. For $4 \wedge 2$ ($0100 \wedge 0010$), the result is 0110 , or 6.

$4 \wedge 2$ is 6

Read problem statement 1 and
2. Read 2 integer values to perform relational operators: $<$, $>$, $<=$, $>=$, $!=$, $==$.

Ans: step 1: Evaluate the relational operators for the values.

The example integer values are

$a = 4$ and $b = 2$.

The relational operators to be performed are

$<$, $>$, $<=$, $>=$, $==$, and $!=$.

* Less than ($<$):

$a < b$

$4 < 2$

This statement is False.

$4 < 2$ is False (0).

* Greater than ($>$):

$a > b$

$4 > 2$

This statement is True.

4 > 2 is True
* Less than or equal to (\leq): 28% 9:42 pm

$$a \leq b$$

$$4 \leq 2$$

This statement is False.

$4 \leq 2$ is False (0).

* Greater than or equal to (\geq):

$$a \geq b$$

$$4 \geq 2$$

This statement is True.

$4 \geq 2$ is True (1)

* Equal to ($=$):

$$a = b$$

$$4 = 2$$

This statement is False.

$4 = 2$ is False (0)

* Not equal to (\neq):

$$a \neq b$$

$$4 \neq 2$$

This statement is True.

$4 \neq 2$ is True (1)

3.

- i) MCQ's on right shift & left shift operation
- ii) MCQ's on Increment & decrement.

ms, (i) 1. what is the primary function of the right shift (\gg) operator for positive integers in most programming languages (C, C++, Java, Python)?

- A) Multiply the number by 2 for each shift.
- ☒ B) Divide the number by 2 for each shift (integer division / floored)
- C) Add 2 to the number for each shift.
- D) Subtract 2 from the number for each shift.

2. what is the result of the expression $10 \gg 1$ in C?

- A) 10
- B) 20
- ☒ C) 5
- D) 1

3. When an unsigned integer is right-shifted, what value are the vacated leftmost bit positions filled with?

- A) The sign bit (most significant bit).
- B) ones (1s)
- ☒ C) zeros (0s)
- D) It depends on the compiler implementation.

4. In C programming, what is the result of the expression $16 \gg 3$?

- A) 0
- ☒ B) 2
- C) 4
- D) 8

5. If an unsigned char variable x holds the value 255 (binary 11111111), what is the value of the expression $x > 54$?

- A) 255
- B) 15
- C) 0
- D) 240

(16) In C, what is the result of the expression $12 < 2$?

- A) 3
- B) 24
- C) 48
- D) 6

(ii)

1. what is the primary function of increment ($++$) operator in C?

- a) Multiplies the value of the variable by 1.
- b) Increments the value of the variable by 1.
- c) Decrements the value of the variable by 1.
- d) Divides the value of the variable by 1.

2. what will be the value of j after the following code snippet executes?

```
C
int i = 5, j;
j = i++;
```

- a) 5
- b) 6
- c) 4
- d) undefined

3. Which of the following is an invalid operand for the increment and decrement operators?

- a) A variable ($i++$)
- b) A pointer ($*p++$)
- c) A constant ($++5$)
- d) A floating-point number ($f++$)

4. What is the output of the following C code?

```
C
int x=5, y;
y=x++/2;
printf("%d", y);
```

- a) 3
- b) 2
- c) 2.5
- d) compile-time error

5. What will be the final value of a and b after the following code snippet executes?

```
C
int a=4, b, c;
b=--a;
c=a--;
```

- a) $a=3, b=3, c=3$
- b) $a=2, b=3, c=3$
- c) $a=2, b=3, c=2$
- d) $a=3, b=2, c=3$