

What is the answer for $x=x|y$ where $x=-5$ and $y=2$? *

2 points

- ☐ 5
- ☐ 2
- ☒ -5
- ☐ -2

what is output of $5 \gg 3$?

2 points

- ☒ 0
- ☐ 1
- ☐ 2
- ☐ 5

what is the output of $64 \gg \gg 4$

2 points

- ☐ 8
- ☐ 12
- ☐ 16
- ☒ 4

Write the code to perform swapping of bits can be possible if values at i and j positions for a no(01001001) i=6 and j to be checked .

2 points

- ☒ $((no \gg i) \& 1) \neq ((no \gg j) \& 1)$
- ☐ $((no \ll i) \& 1) \neq ((no \ll j) \& 1)$
- ☐ $((no \ll i) | 1) \neq ((no \ll j) | 1)$
- ☐ $((no \gg i) \& 1) \neq ((no \ll j) \& 1)$

What is the value of x in each iteration when x=7?

2 points

```
public static short parity(long x) {  
    short result = 0;  
    while (x != 0) {  
        result ^= 1;  
        x &= (x - 1); // Drops the lowest set bit of x.  
    }  
    return result;  
}
```

- ☒ 6 4 1
- ☐ 7 5 3
- ☐ 7 6 5
- ☐ 6 3 1

What is the time complexity for the above code for x=7?

1 point

- ☐ $O(n)$
- ☒ $O(k)$
- ☐ $O(1)$
- ☐ None of these

We illustrate the approach with an 8-bit word. The parity of (11010111) is the same as the parity of (1101) XORed with (0111), i.e., of (1010). Note that the first XOR yields (11011010), and only the last 4 bits are relevant going forward. The second XOR yields (11101100), and only the last 2 bits are relevant. The third XOR yields (10011010). The last bit is the result x , and to extract we have to perform a _____ operation to find the parity?

```
public static short parity(long x) {  
    x ^= x >>> 32;  
    x ^= x >>> 16;  
    x ^= x >>> 8;  
    x ^= x >>> 4;  
    x ^= x >>> 2;  
    x ^= x >>> 1;  
}
```

- ☐ OR
- ☒ AND
- ☐ NOT
- ☐ XOR

What is the time complexity of the above program ?

1 point

O(logn)

Define the weight of a nonnegative integer x to be the number of bits that are set to 1 in its binary representation. For example, since 92 in base-2 equals $(1011100)_2$, the weight of 92 is

- ☐ 3
- ☒ 4
- ☐ 0
- ☐ 1

FIND A CLOSEST INTEGER WITH THE SAME WEIGHT for $x=6(0000\ 0110)$
NUM_UNSIGN_BITS=63

2 points

```
public static long closestIntSameBitCount(long x) {  
    // x is assumed to be non-negative so we know the leading bit is 0. We  
    // restrict to our attention to 63 LSBs.  
    for (int i = 0; i < NUM_UNSIGN_BITS - 1; ++i) {  
        if (((x >> i) & 1) != ((x >> (i + 1)) & 1)) {  
            x ^= (1L << i) | (1L << (i + 1)); // Swaps bit-i and bit-(i + 1).  
            return x;  
        }  
    }  
  
    // Throw error if all bits of x are 0 or 1.  
    throw new IllegalArgumentException("All bits are 0 or 1");  
}
```

- ☐ 7
- ☒ 5
- ☐ 6
- ☐ 4

Find the value of sum where x=5 and y=3?

2 points

```
public static long foo(long x, long y) {  
    long sum = 0;  
    while (x != 0) {  
        // Examines each bit of x.  
        if ((x & 1) != 0) {  
            sum = sum + y;  
        }  
        x >>= 1;  
        y <<= 1;  
    }  
    return sum;  
}
```

- ☐ 5
- ☐ 10
- ☒ 15
- ☐ 20

This form was created inside of SIKSHA 'O' ANUSANDHAN.

Google Forms