



You are working on problem set:
[Homework 2](#) ([Pause](#))



bits_to_flip



Language/Type: **C** [C bitwise operators](#) [bit algorithms](#) [bit manipulation](#)

Related Links:

[Wikipedia: Ones' complement](#)

Write a function named **bits_to_flip** that accepts as parameters two integers *a* and *b* and returns the number of bits that would need to be flipped from 0 to 1 or vice versa to turn *a*'s binary representation into *b*'s. For example, if the call is `bits_to_flip(42, 108)`, the binary representation of 42 is `0101010` preceded by 25 zeros, and the binary representation of 108 is `1101100` preceded by 25 zeros. To convert the first into the second you would need to flip three bits. So your function should return 3.

You should solve this problem using bitwise operators and arithmetic. Do not use any collections or strings to solve the problem. You may assume that an `int` occupies 32 bits in memory.

```
1 // bits_to_flip by Brandon Kmiec. Submitted for CSC152 September 15, 2024
2 // function to determine the number of bits to flip to turn a into b
3
4 int bits_to_flip(int a, int b) {
5     int count = 0;
6     int aXorB = a ^ b;
7
8     for(int i = 0; i < 32; i++) {
9         if((aXorB & (1 << i)) > 0)
10             count++;
11     }
```

```

11     } // end for
12
13     if((a < 0 && b >= 0) || (a >= 0 & b < 0))
14         count++;
15
16     return count;
17 } // end bits_to_flip

```

Function: Write a C function as described, not a complete program.



Submit



✓ You passed 12 of 12 tests.



✓ bits_to_flip(42, 108)	→ 3
✓ bits_to_flip(7, 10)	→ 3
✓ bits_to_flip(32, 31)	→ 6
✓ bits_to_flip(42, 108)	→ 3
✓ bits_to_flip(55, 72)	→ 7
✓ bits_to_flip(12345, 54321)	→ 5
✓ bits_to_flip(0, 0)	→ 0
✓ bits_to_flip(0, 2147483647)	→ 31
✓ bits_to_flip(-1, 1)	→ 31
✓ bits_to_flip(65535, -1)	→ 16
✓ bits_to_flip(-55, -2987653)	→ 12
✓ bits_to_flip(-2147483648, 0)	→ 1

Testing began at 2024/09/14 19:16 (PDT) and ran for 414 ms.



Need help?

Stuck on an exercise? [Contact your TA or instructor](#).

If something seems wrong with our site, please [contact us.](#)