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

cmp_bits

Language/Type: C C bitwise operators bit manipulation
Author: Cynthia Lee (on 2016/11/03)

Write a function named **cmp_bits** that accepts two int arguments a and b, and compares the number of "on" bits (number of 1s) between them. Returns a negative result if the bitwise representation of a has fewer 1s than b, a positive result if a has more 1s than b, and zero if both have the same number of 1s. For example, if a is 5 (binary 00000000000000000000000000000101) and b is 32 (binary 00000000000000000000000000010000), then cmp_bits returns any positive number (for example, it could return 1).

```c
// cmp_bits by Brandon Kmiec. Submitted for CSC 152 September 15, 2024
// function used to compare the number of ON bits between two numbers
// Used https://en.wikipedia.org/wiki/Mask_(computing)#Querying_the_status_of_a_bit to learn how to pick out a single bit

int cmp_bits(int a, int b) {
    if(a == b)
        return 0;

    int aCount = 0;
    int bCount = 0;
    int bitMaskA = 0;
    int bitMaskB = 0;
    int tmp = 1;

    for(int i = 0; i < 31; i++) {
        bitMaskA = a & tmp;
        bitMaskB = b & tmp;

        if(bitMaskA > 0)
            aCount++;

        if(bitMaskB > 0)
            bCount++;

        tmp = tmp << 1;
    } // end for

    if(aCount > bCount)
        return 1;
    else if(aCount < bCount)
        return -1;
    else
        return 0;

} // end cmp_bits
```

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

Submit

You passed 11 of 11 tests.

- cmp_bits(5, 16) → 1
- cmp_bits(16, 5) → -1

cmp_bits(6, 18)	→	0
cmp_bits(8, 2)	→	0
cmp_bits(0, 0)	→	0
cmp_bits(5, 5)	→	0
cmp_bits(5, 10)	→	0
cmp_bits(524288, 5)	→	0
cmp_bits(-1, -1)	→	0
cmp_bits(-1, 0)	→	1
cmp_bits(0, -1)	→	-1

Testing began at 2024/09/14 19:12 (PDT) and ran for 413 ms.