

```

#include <stdio.h>

int main(){
    int a, b, c, p, q, r, c1 = 0, c2 = 0, c3 = 0, c4 = 0;
    int i, j, k, sum, counter, disc;
    int x, X, y, Y, z, Z;
    int isBoundary, isPrime, isReal;

    scanf("(%d,%d,%d)(%d,%d,%d)", &a, &b, &c, &p, &q, &r);

    // Find the smallest cuboid that contains these two points
    // (x,y,z) being one extreme corner and (X,Y,Z) being another
    x = (a < p)? a : p;
    X = a + p - x;
    y = (b < q)? b : q;
    Y = b + q - y;
    z = (c < r)? c : r;
    Z = c + r - z;

    for(i = x; i <= X; i++){
        for(j = y; j <= Y; j++){
            for(k = z; k <= Z; k++){
                // Is this a boundary point?
                isBoundary = 0;
                if(i == 1 || i == 256)
                    isBoundary = 1;
                if(j == 1 || j == 256)
                    isBoundary = 1;
                if(k == 1 || k == 256)
                    isBoundary = 1;

                // Is the sum of coordinates prime?
                sum = i + j + k;
                isPrime = 1;
                for(counter = 2; counter < sum; counter++)
                    if(sum % counter == 0)
                        isPrime = 0; // Not a prime sum

                // Does this equation have real roots?
                isReal = 1;
                disc = j * j - 4 * i * k;
                if(disc < 0)
                    isReal = 0; // Complex roots

                // Apply the conditions in correct order
                // as specified in the question
                if(isBoundary)
                    c1++;
                else if(isPrime)
                    c2++;
                else if(isReal)
                    c3++;
                else
                    c4++;
            }
        }
    }

    printf("%d\n%d\n%d\n%d\n%d", c1, c2, c3, c4, c1+c2+c3-c4);

    return 0;
}

```