* Let ⌊a/b⌋=a mod b=k.
* Which means a given k as remainder when divided by b and a gives k as quotient when divided by b
* By euclidean lemma , we have a = k\*b + k where 0< k < b
* Since b > k

=> k\*b > k\*k multiply both sides by k

=> k\*b + k > k\*k add k>0 to LHS

=> a > k\*k from above equation

=> x > k\*k max(a) = x

So, we can assume k<=sqrt(x)

* We can loop over k and find pairs a,b satisfying special property with k fixed
* Now, For any fixed k , 1<=k<=sqrt(x)

We have b > k and 1<=b<=y and 1<=a<=x

Substitute a=b\*k+k then we have,

1<=k\*b+k<=x

=> 1/k<=b+1<=x/k divide by k>0

=> 1/k-1<=b<=x/k -1 subtract 1

Combining this with above equation we get,

k<b<=x/k-1

* k<b<=x/k-1
* So, for any fixed k >0 , the number of pairs satisfying condition is

max( 0 , min( y , x/k-1 ) - k )

*def* count\_special\_pairs(*x*, *y*):

count = 0 # initialize count

# loop for all values from 1 to sqrt(x)

for k in range(1, int(math.sqrt(*x*)) + 1):

# max(0, min(y, x // k - 1) - k) is the number of values of b in the range [k+1, min(y, x // k - 1)] that satisfy a % b = a // b

count += max(0, min(*y*, *x* // k - 1) - k)

return count # return the count