#include <cstdio>

#include <algorithm>

using namespace std;

int n;

long long a, b;

bool tx[200010], ty[200010];

int x[200010], y[200010], p[200010][2];

int \*gx[200010], \*gy[200010], dx[200010], dy[200010];

int xi(int px)

{

int s=0, e=n-1, k;

while (e-s > 1)

{

k = (s+e)>>1;

if (x[k] == px)

return k;

if (x[k] > px)

e = k;

else

s = k;

}

if (x[s] == px)

return s;

return e;

}

int yi(int py)

{

int s=0, e=n-1, k;

while (e-s > 1)

{

k = (s+e)>>1;

if (y[k] == py)

return k;

if (y[k] > py)

e = k;

else

s = k;

}

if (y[s] == py)

return s;

return e;

}

void rec(int l, bool type)

{

int i;

if (type == 0)

{

for (i = 0; i < dx[l]; i++)

if (!ty[gx[l][i]])

{

b++;

ty[gx[l][i]] = 1;

rec(gx[l][i], 1);

}

}

else

for (i = 0; i < dy[l]; i++)

if (!tx[gy[l][i]])

{

a++;

tx[gy[l][i]] = 1;

rec(gy[l][i], 0);

}

}

int main()

{

int k, l, i;

long long m;

scanf("%d", &n);

for (i = 0; i < n; i++)

{

scanf("%d%d", x+i, y+i);

p[i][0] = x[i];

p[i][1] = y[i];

}

sort(x, x+n);

sort(y, y+n);

for (i = 0; i < n; i++)

{

dx[xi(p[i][0])]++;

dy[yi(p[i][1])]++;

}

for (i = 0; i < n; i++)

{

gx[i] = new int[dx[i]+1];

gy[i] = new int[dy[i]+1];

dx[i] = dy[i] = 0;

}

for (i = 0; i < n; i++)

{

k = xi(p[i][0]);

l = yi(p[i][1]);

gx[k][dx[k]++] = l;

gy[l][dy[l]++] = k;

}

m = 0;

for (i = 0; i < n; i++)

if (!tx[i])

{

a = 1;

b = 0;

tx[i] = 1;

rec(i, 0);

m += a\*b;

}

printf("%lld\n", m);

return 0;

}