

E. Bus Number

time limit per test 1 second
 memory limit per test 256 megabytes
 input standard input
 output standard output

This night wasn't easy on Vasya. His favorite team lost, and he didn't find himself victorious either — although he played perfectly, his teammates let him down every time. He had to win at least one more time, but the losestreak only grew longer and longer... It's no wonder he didn't get any sleep this night at all.

In the morning, Vasya was waiting the bus to the university on the bus stop. Vasya's thoughts were hazy and so he couldn't remember the right bus' number quite right and got onto the bus with the number n .

In the bus, Vasya thought that he could get the order of the digits in the number of the bus wrong. Furthermore, he could "see" some digits several times, but the digits he saw were definitely in the real number of the bus. For example, if Vasya saw the number 2028, it could mean that the real bus number could be 2028, 8022, 2820 or just 820. However, numbers 80, 22208, 52 definitely couldn't be the number of the bus. Also, real bus number couldn't start with the digit 0, this meaning that, for example, number 082 couldn't be the real bus number too.

Given n , determine the total number of possible bus number variants.

Input

The first line contains one integer n ($1 \leq n \leq 10^{18}$) — the number of the bus that was seen by Vasya. It is guaranteed that this number does not start with 0.

Output

Output a single integer — the amount of possible variants of the real bus number.

Examples

| | |
|---------------|-------------|
| input | Copy |
| 97 | |
| output | Copy |
| 2 | |

| | |
|---------------|-------------|
| input | Copy |
| 2028 | |
| output | Copy |
| 13 | |

Note

In the first sample, only variants 97 and 79 are possible.

In the second sample, the variants (in the increasing order) are the following: 208, 280, 802, 820, 2028, 2082, 2208, 2280, 2802, 2820, 8022, 8202, 8220.

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  #define ll long long
4  #define vi vector<int>
5  int cnt[10],k;
6  ll fac[20],ans=0;
7  void fun(int pos, vi v)
8  {
9      if(pos==10){
10         ll ret = 0;
11         int tot = 0;
12         for(int i=0; i<v.size(); i++) tot += v[i];
13         ret = fac[tot];
14         for(int i=0; i<v.size(); i++) ret /= fac[v[i]];
15         if(v[0]>0){
16             ll temp = fac[tot-1];
17             temp /= fac[v[0]-1];
18             for(int i=1; i<v.size(); i++) temp /= fac[v[i]];
19             ret -= temp;
20         }
21         ans += ret;
22         return;
23     }
24     if(cnt[pos]==0){
25         vi u = v;
26         u.push_back(0);
27         fun(pos+1,u);
28     }
29     else{
30         for(int i=1; i<=cnt[pos]; i++){
31             vi u = v;
32             u.push_back(i);
33             fun(pos+1,u);
34         }
35     }
36 }
37
38 int main()
39 {
40     fac[0] = 1;
41     for(ll i=1; i<=20; i++) fac[i] = fac[i-1]*i;
42
43     ll n; cin>>n;
44
45     while(n!=0){
46         int r = n%10LL;
47         cnt[r]++;
48         n /= 10LL;
49     }
50
51     ans = 0;
52     vi v;
53     fun(0,v);
54     cout << ans << endl;
55
56     return 0;
57 }
```