1049. Counting Ones (30)

时间限制

10 ms

内存限制

65536 kB

代码长度限制

16000 B

判题程序

Standard

作者

CHEN, Yue

The task is simple: given any positive integer N, you are supposed to count the total number of 1's in the decimal form of the integers from 1 to N. For example, given N being 12, there are five 1's in 1, 10, 11, and 12.

**Input Specification:**

Each input file contains one test case which gives the positive N (<=230).

**Output Specification:**

For each test case, print the number of 1's in one line.

**Sample Input:**

12

**Sample Output:**

5

原理：对于每一个数位进行统计：

若该数位为1，则对于该数位的高位的任何一个数字（除去其本身）来说{set:高位数字确定，则对于这个确定的高位数字，有当前数位的位数（例如，在十位上，则位数为十）个不同的表达方式（exception：该数位的更高位其本身，例如，1123，其中统计：（0,1,2…10都可以有10^2个表达方式），但是12不行）}，然后，对于它的低位数，即截取包括这个数位的所有更高位剩下的数字，这个数字加1，是高位数取其本身时所代表的不同表达数量；

其他情况：0，1+,以此类推。

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

#include<vector>

#include<sstream>

#include<stack>

using namespace std;

int amount = 0;

void cipher(int num)

{

int base = 1;

int low,cur,high;

while (1)

{

if (num%base == num)

{

if (base != 1)

break;

}

cur = (num / base) % 10;

high = num / (base \* 10);

low = num - num / base\*base;

if (cur == 1)

{

amount += high\*base + low + 1;

}

else if (cur == 0)

{

amount += high\*base;

}

else

{

amount += (high + 1)\*base;

}

base \*= 10;

}

}

int main()

{

int num;

cin >> num;

cipher(num);

cout << amount;

}