

数位 dp

- 简介：
 - 关于问题：常常是某整数区间内，的一些计数问题；比如，满足区间内满足某条件的整数个数。
 - 由于是连续的区间的数。所以小区间的统计和大区间的统计之间存在关系。

比如说
10000.....19999
0....9999这段后缀的变化是相似的。
意味着。
如果关注的数，就是后缀中，10不相连之类的。前面满足的情况，前面较小的数字区间也有一部分可以被满足。

记忆化搜索简单的模板

```
#include <iostream>
#include <algorithm>
using namespace std;

int f[20][20][2]; //记忆化搜索。后面limit也记录，优化，减少if特判。且空间复杂度肯定是没有问题的
int a[20];          //用来记录当前数上的各个位置上的一些信息。

int dfs(int no /*第几位*/, int now /*题意关注的信息*/, int limit) // limit:
//用来表示前缀是否贴着上界。
{
    memset(f, -1, sizeof(f));
    if (!no) //是否返回看具体题意。如果一直保证合法合法搜索得到的前缀，就返回1.否则返回判断。
        return 1;
    int mmax = limit ? a[no] : 9, res = 0; //确认合法搜索的取值空间。如果贴着上界，要保证枚举数就在范围之内。
    for (int i = 0; i <= mmax; i++)
        res += dfs(no - 1, i, a[no] == i & limit);
    return f[no][now][limit] = res;
}

int solve(int x)
{
    int len = 0;
    while (x)
        a[++len] = x % 10, x /= 10;
    return dfs(len, a[len], true);
}
```

}