

动态规划 1

数位DP

算法简介

一般用于处理形如区间 $[l, r]$ 中满足条件的数有几个之类问题。

算法思想

首先考虑将区间 $[l, r]$ 拆分成 $[0, r] - [0, l]$

考虑记忆化搜索，一般搜索函数为 $f(pos, s, eq, zero)$

其中 pos 表示当前搜索位置 s 表示前缀状态 eq 表示前缀是否与查询上界相等 $zero$ 表示是否有前导零。

算法练习

[洛谷p2657](#)

题意

询问区间 $[l, r]$ 中满足相邻数位之差至少为 2 的数的个数。

题解

```
int a[10], dp[10][10];
int dfs(int pos, int pre, bool eq, bool zero) {
    if (!pos) return 1;
    if (!eq && !dp[pos][pre]) return dp[pos][pre];
    int ans = 0, v = eq ? a[pos] : 9;
    for (int i = 0; i < v; ++i) {
        if (!zero && abs(i - pre) < 2)
            continue;
        ans += dfs(pos - 1, i, eq && i == a[pos], zero && i == 0);
    }
    if (!eq && !zero) dp[pos][pre] = ans;
    return ans;
}
int solve(int n) {
    int len = 0;
    mem(dp, -1);
```

```
while(n){
    a[++len]=n%10;
    n/=10;
}
int main()
{
    int a=read_int(),b=read_int();
    printf("%d",solve(b)-solve(a-1));
    return 0;
}
```

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:jxm2001:%E5%8A%A8%E6%80%81%E8%A7%84%E5%88%92_1&rev=1596109786

Last update: 2020/07/30 19:49

