

박성원

<https://www.acmicpc.net/problem/1086>

- 15개 이하의 숫자 집합을 통하여 하나의 숫자를 만드므로 최대 15!의 경우의 수가 나온다.
15!은 매우 큰 값이다. 따라서 나머지 연산을 이용하여 수를 줄인다.

- 나머지 연산 :

① $(A+B) \% C = (A\%C + B\%C) \% C$

② $(A * B) \% C = (A\%C * B\%C) \% C$

- $dp[i][j]$ = 사용한 순열에 포함된 수의 집합이 i 이고 나머지가 j 인 것의 개수
- i 에 포함되어 있지 않은 수를 k 라고 했을 때

① k 번째 수 : $A[k]$

② k 번째 수가 포함된 경우의 나머지 = $(j * 10^{\text{자릿수}} + A[k]) \% M$

집합 i

$$= ((j * 10^{\text{자릿수}} \% M) + (A[k] \% M)) \% M$$



$A[k]$ 길이 만큼 자릿수가 증가하고 끝자리는 $A[k]$ 가 된다.

- $dp[i | (1 \ll k)][next] += dp[i][j]$

박성원

<https://www.acmicpc.net/problem/1086>

```
dp[0][0] = 1;
for (int i = 0; i < (1 << N); ++i) {
    for (int j = 0; j < M; ++j) {
        for (int k = 0; k < N; ++k) {
            if ((i & (1 << k)) == 0) {
                int next = j * digit[snum[k].size()];
                next %= M;
                next += A[k];
                next %= M;
                dp[i | (1 << k)][next] += dp[i][j];
            }
        }
    }
}
```