

[백준] 1256번: 사전

통계 수정 삭제

yooool · 방금 전

 0

2023_Autumn_Algorithm_Study



▼ 목록 보기

5/5



1256번: 사전

사전에 N개의 "a"와 M개의 "z"로 이루어진 문자열을 알파벳 순서대로 만들고, K번째 문자열을 출력한다.

입력: N, M, K

출력: K번째 문자열(사전에 수록된 문자열의 개수가 K보다 작으면 -1 출력)

 $1 \leq N, M \leq 100$ $1 \leq K \leq 1,000,000,000$

combination 함수를 통해 만들 수 있는 문자열의 개수를 계산했다.

aorz 함수를 반복적으로 호출하며 문자열의 왼쪽부터 dic에 저장할 수 있도록 했다.

N = 3, M = 2일 때 가능한 조합의 수는 10이다.

이때 가장 왼쪽 알파벳이 a라면 가능한 조합의 수는 6으로 줄게 된다.

z일 경우 조합의 수는 4가 된다.

남은 조합들로만 다시 aorz 함수에 넣게 되면 문자열을 구할 수 있다.

factorial 함수에 자꾸 음수가 들어가 오류가 발생해 aorz 함수에서 comb의 값을 범위에 따라 다르게 설정했다.

vs에서는 잘 돌아갔는데 백준에 넣으면 자꾸 런타임 에러(IntegerOverflow)가 발생했다. 이건 해결이 안된다.....

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
```

```

#include <stdlib.h>
#define INF 0x3f3f3f3f3f3f3f3f

int cnt;
char* dic;

long long factorial(int N) {
    if (N == 1 || N == 0) return 1;
    long long fac = N * factorial(N - 1);
    if (fac > 10000000000) fac = INF;
    return fac;
}

long long combination(int N, int M) {
    long long com = factorial(N + M) / (factorial(N) * factorial(M));
    if (com > 10000000000) com = INF;
    return com;
}

void aorz(int N, int M, long long K, int i) {

    long long comb;

    if (i == cnt) return;

    if (N <= 1 && M != 1)
        comb = M - 1;
    else if (N != 1 && M <= 1)
        comb = N - 1;
    else if (N == 1 && M == 1)
        comb = 1;
    else
        comb = combination(N - 1, M);

    if (comb >= K) {
        dic[i] = 'a';
        N--;
    }
    else {
        dic[i] = 'z';
        M--;
        K -= comb;
    }

    aorz(N, M, K, i + 1);
}

int main() {

    int N, M, i = 0;
    long long K;

    scanf("%d %d %lld", &N, &M, &K);

    if (K > 10000000000) return 0;

    cnt = N + M;

    if (K > combination(N, M)) {
        printf("-1");
        return 0;
    }
}

```

```
dic = (char*)malloc((cnt + 1) * sizeof(char));

aorz(N, M, K, i);

for (int j = 0; j < cnt; j++)
    printf("%c", dic[j]);

free(dic);

return 0;
}``
```



김지울



이전 포스트

[백준] 11650번: 좌표 정렬하기

0개의 댓글

댓글을 작성하세요

댓글 작성



Powered by
Stellate