

2023_Autumn_Algorithm_Study_5주차 과제 기록

통계 수정 삭제

jaesung2da · 방금 전 · 비공개

❤ 0

5주차 과제는
1256번: 사전

사전



시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	128 MB	16115	4868	3566	30.920%

문제

동호와 규완이는 212호에서 문자열에 대해 공부하고 있다. 김진영 조교는 동호와 규완이에게 특별 과제를 주었다. 특별 과제는 특별한 문자열로 이루어진 사전을 만드는 것이다. 사전에 수록되어 있는 모든 문자열은 N개의 "a"와 M개의 "z"로 이루어져 있다. 그리고 다른 문자는 없다. 사전에는 알파벳 순서대로 수록되어 있다.

규완이는 사전을 완성했지만, 동호는 사전을 완성하지 못했다. 동호는 자신의 과제를 끝내기 위해서 규완이의 사전을 몰래 참조하기로 했다. 동호는 규완이가 자리를 비운 사이에 몰래 사전을 보려고 하기 때문에, 문자열 하나만 찾을 여유밖에 없다.

N과 M이 주어졌을 때, 규완이의 사전에서 K번째 문자열이 무엇인지 구하는 프로그램을 작성하시오.

입력

첫째 줄에 세 정수 N, M, K가 순서대로 주어진다.

출력

첫째 줄에 규완이의 사전에서 K번째 문자열을 출력한다. 만약 규완이의 사전에 수록되어 있는 문자열의 개수가 K보다 작으면 -1을 출력한다.

제한

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

예제 입력 1 복사

2 2 2

예제 출력 1 복사

azaz

예제 입력 2 복사

2 2 6

예제 출력 2 복사

zzaa

예제 입력 3 복사

10 10 1000000000

예제 출력 3 복사

-1

예제 입력 4 복사

7 4 47

예제 출력 4 복사

aaazazaazaz

개념공부

사전

우선 사전이라는 것을 처음 접해봐서 구글에 '알고리즘 사전'을 검색해보았다. [참조 블로그](#) 와 근데 이 블로그 쓰신 분도 세종대생인 것 같다. 사용한 이미지 파일이 누가봐도 세종대 교안이랑 똑같다. 사전 ADT는 주로 연락처 목록, 신용카드 사용승인, 인터넷 주소 매핑 등에 활용될 수 있고 간접적으로는 알고리즘 수행을 위한 보조 데이터구조로 많이 활용된다.

사전은 리스트, 트리, 해시테이블을 이용하여 구현 될 수 있다고 한다. 그 중에서도 블로그에서는

리스트와 관련된 무순 사전 ADT와 순서 사전 ADT만 있어서 이것들을 먼저 공부해보려고한다. 아마 이분도 우리학교 교안으로 보니까 학교에서도 이 두개를 먼저하거나 두개만 하지 않을까..?

무순 사전 ADT: 기록파일

무순 사전 ADT는 기존 리스트에 새로운 원소가 삽입될 때, 맨 앞 또는 맨 뒤에 삽입하면 되기 때문에 $O(1)$ 의 시간이다.

리스트에 원소가 저장되는 기준이 없으므로, 탐색이나 삭제에서는 최악의 경우 $O(n)$ 시간이 걸린다.

무순 사전ADT가 효율적인 경우는 소규모의 사전이나. 서버의 로그인 기록.

무순 사전 ADT는 선형탐색을 사용하고 그냥 리스트를 순회하면서 Key값을 확인하는 것이다.

순서 사전 ADT: 일람표

순서리스트가 사용되어 구현된 사전이다. key를 기준으로 이미 정렬이 되어 있어 탐색에 용이하다.

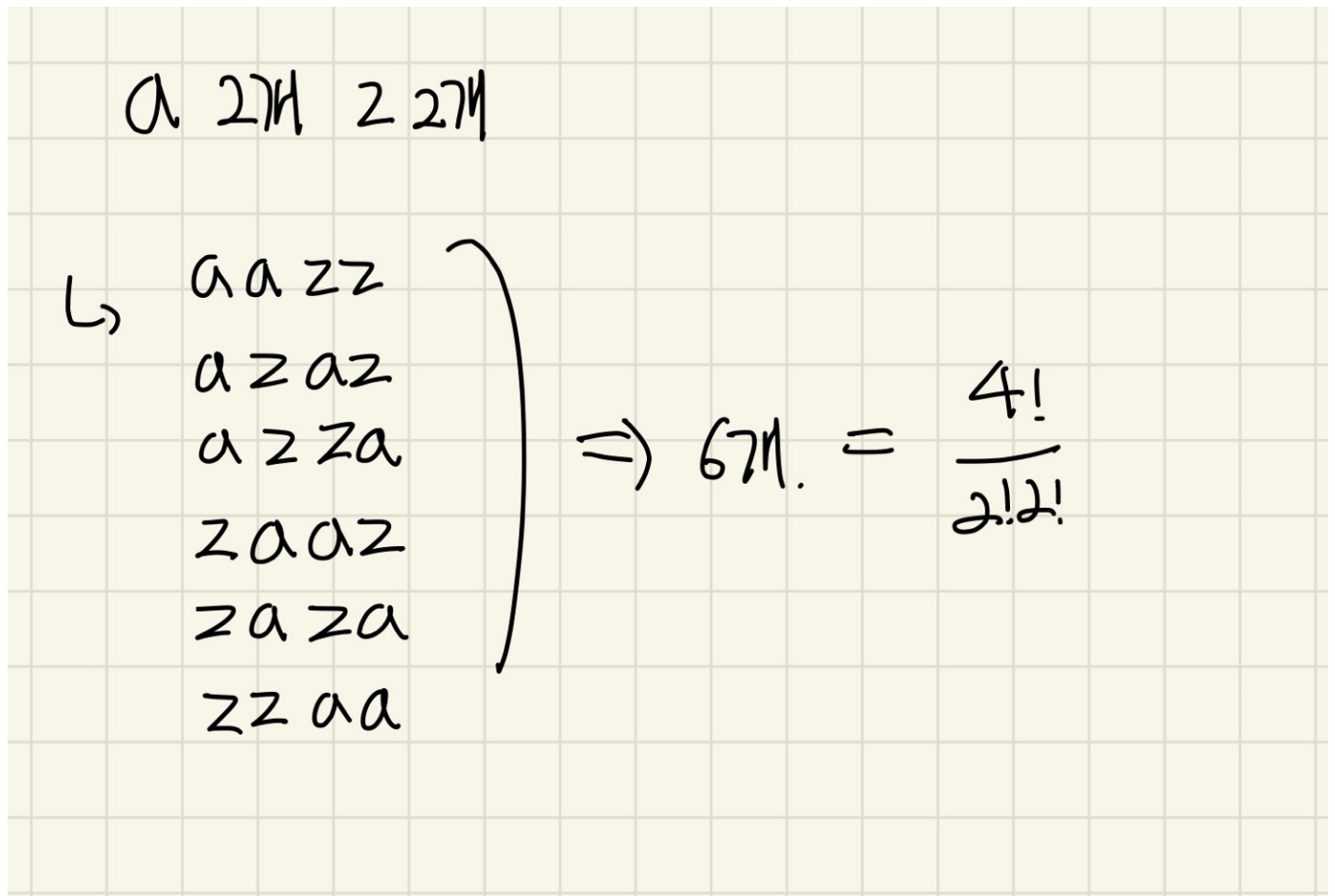
하지만 삽입하는 경우에 정렬 기준에 맞는 위치에 삽입해야하므로 무순사전 ADT에서는 $O(1)$ 이었던 시간이 순서사전 ADT에서는 최악의 경우 $O(n)$ 의 시간이 필요하다.

즉 순서사전이 용이한 경우는 탐색을 자주하는 경우이다.

순서사전 ADT는 이진탐색을 사용한다 이진탐색은 배열이 있다고 했을 때 그 배열의 가운데를 기준으로 크다 작다로 나눈다. UP DOWN 게임을 생각해서 원하는 것을 찾기 위해 항상 가운데를 기준으로 찾아가는 방식이라고 한다.

문제 풀기 전 전략

일단 공부를 하긴 했는데 문제에 어떻게 적용해야 할지는 감이 안잡힌다. 내가 문제를 읽고 알 수 있는 것은 그저 동호가 나쁜 새끼라는 거다.



아무리 생각해도 문자열 개수가까지는 알겠는데.. 이걸 어떻게 사전을 이용하지.. 그냥 if문 엄청 써서 풀어야할 것 같은데...

N	M	K
11	11	4
3	2	2

1) a a a z z

2) a a z a z

3) a a z z a

4) a z a a z

5) a z a z a

6) a z z a a

7) z a a a z

8) z a a z a

9) z a z a a

10) z z a a a

$$\Rightarrow \frac{2!}{1!1!}$$

$$\Rightarrow \frac{3!}{2!1!} \frac{(N+M-2)!}{N-2}$$

$$\Rightarrow \frac{4!}{2!2!} \frac{(N+M-1)!}{(N-1)! m!}$$

내 지능의 문제인지 아무리 생각해도 모르겠..

문제점 그리고 해결

```
#pragma warning(disable:4996)
#include <stdio.h>
```

```

int fac(int A) {

    int result = 1;

    for(int i = A; i > 0; i--) result = result * i;

    return result;
}

int line(int N, int M) {

    int result;

    result = fac(N + M - 1) / (fac(N)*fac(M));

    return result;
}

int main() {

    int N, M, K;

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

    int fac_NM = fac(N + M), fac_N=fac(N), fac_M = fac(M);

    int cnt = fac_NM / (fac_N * fac_M);

    int* arr = (int*)malloc(sizeof(int) * cnt);

    if (K > cnt) printf("-1");
    else {
        for (int i = 0; i < cnt; i++) {

            if (K < line(N, M)) {
                arr[i - 1] = 'a';
                N--;
            }
            else {
                arr[i - 1] = 'z';
                M--;
            }
        }
    }

    for (int i = 0; i < cnt; i++) {
        printf("%c",arr[i]);
    }

    return 0;
}

```

문제가 아주 많다 애초에 답이 안나온다 zzzz만 나온다. 될리가 없지..

여러 블로그를 참고해도 내가 이해하지 못한 방식을 이용해서 푸는 건 아닌 것 같아서

N	M	K
11	11	4
3	2	2

1) $aaa\ z\ z$

2) $aa\ z\ a\ z$

3) $aa\ z\ z\ a$

4) $a\ z\ a\ a\ z$

5) $a\ z\ a\ z\ a$

6) $a\ z\ z\ a\ a$

7) $z\ a\ a\ a\ z$

8) $z\ a\ a\ z\ a$

9) $z\ a\ z\ a\ a$

10) $z\ z\ a\ a\ a$

$\Rightarrow \frac{2!}{1!1!}$

$\Rightarrow \frac{3!}{2!1!} \quad \frac{(N+M-2)!}{N-2}$

$\Rightarrow \frac{4!}{2!2!} \quad \frac{(N+M-1)!}{(N-1)! \cdot M!}$

이 그림을 토대로 다시 생각해보았다. 기준선을 기준으로 조합의 수보다 작거나 같으면 a가 들어가고 크면 z가 입력된다. 그리고 그 기준을 항상 a로 잡는다고 생각한 다음 코드를 짰다. 중간중간 오류를 잡기 위해 gpt형님의 도움을 받긴 했으나 내가 생각한 로직대로 짜보았더니

```
#pragma warning(disable:4996)
#include <stdio.h>
```

```

#include <stdlib.h>
#include <string.h>

long long nCr(int n, int r);
void append(char *str, int n, int m, long long k);

int main() {
    int N, M;
    long long K;

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

    char *result = (char *)malloc((N + M + 1) * sizeof(char));

    // K가 범위를 넘어서면 -1 출력하고 종료
    if (K < 1 || K > nCr(N + M, N)) {
        printf("-1\n");
        free(result);
        return 0;
    }

    append(result, N, M, K);

    printf("%s\n", result);

    free(result);

    return 0;
}

//nCr = n! / (r! * (n - r)!)
long long nCr(int n, int r) {
    long long result = 1;
    for (int i = 1; i <= r; i++) {
        result *= n - i + 1; //팩토리얼 구현
        result /= i; // 팩토리얼 구현
    }
    return result;
}

void append(char *str, int n, int m, long long k) {
    int cnt_a = n;
    int cnt_z = m;

    for (int i = 0; i < n + m; i++) {
        //a가 아직 존재하면 조합의 수를 반환하고 a가 없다면 0을 반환해서 z가 입력될 수 있게끔한다.
        long long comb = (cnt_a > 0) ? nCr(cnt_a + cnt_z - 1, cnt_a - 1) : 0;

        if (k <= comb) { // k가 조합의 수 즉 a를 기준으로 a가 찍힐 수 있는 조합의 수가 나오면 a를 넣고
            str[i] = 'a';
            cnt_a--;
        }
        else { // k가 a가 나올 수 있는 조합의 수를 넘겼다면 z를 넣는다
            str[i] = 'z';
            cnt_z--;
            k -= comb;
        }
    }
}

```



```

str[n + m] = '\0'; // 쓰레기값이 떠서 NULL값 추가
}

```

위 코드가 나왔고 무려 입출력 값도

```

C:\WINDOWS\system32\cmd.exe
2 2 2
azaz
계속하려면 아무 키나 누르십시오 . . . |

```

```

C:\WINDOWS\system32\cmd.exe
2 2 6
zzaa
계속하려면 아무 키나 누르십시오 . . . |

```

```

C:\WINDOWS\system32\cmd.exe
10 10 1000000000
-1
계속하려면 아무 키나 누르십시오 . . . |

```



나 이정도면 열심히 했잖아

대략 10시 전에 시작했는데 지금 1시58분이니까 못해도 5시간은 했,,,,,,,,,,,,,
난 잘래.. 내일 1교시인데 하ㅏㅏㅏㅏㅏㅏㅏㅏㅏㅏ 미치겠다.



2023 Autumn Algorithm Study 4주차 과제 기록

0개의 댓글

댓글을 작성하세요

댓글 작성

