

beberiche · 32분 전

통계 수정 삭제


DFS


algorithms

깊이우선탐색


백준

이분탐색




0



algorithms



▼ 목록 보기

11/12  

문제
풀이과정
정답

문제

ZOAC

성공



시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	1402	713	576	53.432%

문제

2018년 12월, 처음 시작하게 된 ZOAC의 오프닝을 맡은 성우는 누구보다 화려하게 ZOAC를 알리려 한다.

앞 글자부터 하나씩 보여주는 방식은 너무 식상하다고 생각한 성우는 문자열을 보여주는 새로운 규칙을 고안해냈다!

규칙은 이렇하다. 아직 보여주지 않은 문자 중 추가했을 때의 문자열이 사전 순으로 가장 앞에 오도록 하는 문자를 보여주는 것이다.

예를 들어 ZOAC를 보여주고 싶다면, $A \rightarrow AC \rightarrow OAC \rightarrow ZOAC$ 순으로 보여주면 된다.

바쁜 성우를 위하여 이 규칙대로 출력해주는 프로그램을 작성하시오.

입력

첫 번째 줄에 알파벳 대문자로 구성된 문자열이 주어진다. 문자열의 길이는 최대 100자이다.

출력

규칙에 맞게 순서대로 문자열을 출력한다.

예제 입력 1 복사

ZOAC

예제 출력 1 복사

A
AC
OAC
ZOAC

예제 입력 2 복사

BAC

예제 출력 2 복사

A
AC
BAC

예제 입력 3 복사

STARTLINK

예제 출력 3 복사

A
AI
AIK
AINK
ALINK
ARLINK
ARTLINK
SARTLINK
STARTLINK

풀이과정

재귀문제.

문자열의 순서 규칙을 찾아야 한다. 하나씩 늘어나는 문자열을 사전순으로 나열해야한다.

규칙은 다음과 같다.

1. 시작할 문자, 즉 문자열 가운데 가장 빠른 문자 하나를 찾는다.

2. 해당 문자를 기준으로, 앞과 뒤를 보는 이분탐색을 실행한다.

3. 기존 문자 뒤에 올 수 있는 문자 부터 탐색한다. 기존 문자 뒤에 붙는 문자열이, 기존 문자 앞에 있는 문자열보다 사전순으로 앞이기 때문이다.

4. 따라서 문자열은 앞에 생길 수도, 뒤에 생길 수도 있다. 그러므로, 방문체크를 통해 계속 문자열을 반환해주자 (한번 사용한 문자는 또 쓰지 않으며, 동일한 문자가 2개,3개 나올 수도 있기 때문이다.)

정답

```
public class Main {
```

```

static char[] text;
static boolean[] visited;
public static void main(String[] args) throws Exception {
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

    text = br.readLine().toCharArray();
    visited = new boolean[text.length];
    DFS(0, text.length);
}

private static void DFS(int st, int ed) {
    if(st >= ed) return;

    char[] copyArr = Arrays.copyOfRange(text, st, ed);
    Arrays.sort(copyArr);
    int idx = -1;
    for(int i=st; i<ed; i++) {
        if(text[i] == copyArr[0] && !visited[i]) {
            idx = i;
            break;
        }
    }
    if(idx == -1) return;

    visited[idx] = true;
    String ans = "";
    for(int i=0; i<text.length; i++) {
        if(visited[i]) ans += text[i];
    }
    System.out.println(ans);
    DFS(idx+1, ed);
    DFS(st, idx);
}
}

```



벨

새로운 것에 관심이 많고, 프로젝트 설계 및 최적화를 좋아합니다.



이전 포스트

BOJ 16929 : Two Dots

다음 포스트



BOJ 1700 : 멀티탭 스케줄링

0개의 댓글