1406 에디터

2019-07-02

이승진

<https://www.acmicpc.net/problem/1406>

태그: 구현력

문자 삭제, 삽입 작업을 문자열 연산으로 구현하거나

배열 연산으로 구현하면 O(N) 이라서 시간 초과 오류가 발생한다.

문자열을 링크드 리스트에 넣은 후 (각각의 문자를 링크드 리스트의 노드에)

여기서 삭제, 삽입을 구현하면 O(1) 이다.

Java의 LinkedList 클래스는 double linked list 자료구조이다.

이 클래스를 사용하여 구현하자.

|  |
| --- |
| package baekjoon.p01.p1406;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.util.LinkedList;  import java.util.ListIterator;  import java.util.StringTokenizer;  public class Main {  public static void main(String[] args) throws NumberFormatException, IOException {  BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));  String s = reader.readLine();  LinkedList<Character> list = new LinkedList<>();  for (char c : s.toCharArray()) // 링크드 리스트에 문자 하나씩 넣는다.  list.add(c);  int N = Integer.parseInt(reader.readLine());  ListIterator<Character> iterator = list.listIterator(list.size());  for (int i = 0; i < N; ++i) {  StringTokenizer tokenizer = new StringTokenizer(reader.readLine());  String cmd = tokenizer.nextToken();  switch (cmd) {  case "L": if (iterator.hasPrevious()) iterator.previous(); break; // 완쪽 이동  case "D": if (iterator.hasNext()) iterator.next(); break; // 오른쪽 이동  case "B":  if (iterator.hasPrevious()) {  iterator.previous(); // 먼저 왼쪽으로 이동 후  iterator.remove(); // 오른쪽 항목 삭제  }  break;  case "P":  iterator.add(tokenizer.nextToken().charAt(0)); // 오른쪽에 문자 추가  break;  }  }  StringBuilder builder = new StringBuilder();  for (char c : list)  builder.append(c);  System.out.println(builder.toString());  }  } |

LinkedList 클래스와 ListIterator 사용법만 알면 매우 쉬운 구현이다.