

# 보이어-무어법

문자열 매칭

## 정의

- 패턴의 마지막 문자부터 앞쪽으로 검사를 진행하면서 일치하지 않는 문자가 있으면 미리 준비한 표에 따라 패턴을 옮길 크기를 정한다.
- 성능이나 효과면에서 KMP법보다 더 뛰어 나기 때문에 실무에서도 많이 사용 되는 알고리즘

## 시간 복잡도

- 최악의 경우  $O(n)$ , 평균적으로  $O(n/m)$ 으로 매우 빠르다.

```
// 보이어-무어법으로 문자열 검색
import java.util.Scanner;

class BMmatch {
    //--- 보이어-무어법으로 문자열 검색 ---//
    static int bmMatch(String txt, String pat) {
        int pt; // txt를 따라가는 커서
        int pp; // pat를 따라가는 커서
        int txtLen = txt.length(); // txt의 문자 개수
        int patLen = pat.length(); // pat의 문자 개수
        int[] skip = new int[Character.MAX_VALUE + 1]; // 건너뛰기 표(skip 테이블)

        // skip 테이블 작성
        for (pt = 0; pt ≤ Character.MAX_VALUE; pt++)
            skip[pt] = patLen;
        for (pt = 0; pt < patLen - 1; pt++)
            skip[pat.charAt(pt)] = patLen - pt - 1; // pat = patLen - 1

        // 검색
        while (pt < txtLen) {
            pp = patLen - 1; // pat의 마지막 문자에 주목

            while (txt.charAt(pt) == pat.charAt(pp)) {
                if (pp == 0)
                    return pt; // 검색 성공
                pp--;
                pt--;
            }
            pt += (skip[txt.charAt(pt)] > patLen - pp) ? skip[txt.charAt(pt)] : patLen - pp;
        }
        return -1; // 검색 실패
    }
}
```