Use the Boyer-Moore algorithm to write a function that finds the last occurrence of a substring in a given string and returns its index. Explain why this algorithm can outperform others in certain scenarios.

```
import java.util.Arrays;

public class BoyerMooreDemo {

        private final int R;//radix
        private int[] right;//bad character skip array
        private char[] pattern;// to store the pattern
        private String pat;//string

        //pattern provided as a string
        public BoyerMooreDemo(String pat) {
                this.R=256;
                this.pat=pat;
                right=new int[R];
                Arrays.fill(right,-1);
                for(int j=0;j<pat.length();j++) {
                        right[pat.charAt(j)]=j;
                }//for
        }//end of constructor


        //pattern provided as a character array
        public BoyerMooreDemo(int R, char[] pattern) {
                this.R=R;
                this.pattern=Arrays.copyOf(pattern,pattern.length);
                right=new int[R];
                Arrays.fill(right,-1);
                for(int j=0;j<pat.length();j++) {
                        right[pattern[j]]=j;
                }//for
        }//end of constructor

        public int search(String txt) {
                int M=pat.length();
                int N=txt.length();
                int skip;
                for(int i=0;i<=N-M;i+=skip) {
                        skip=0;
                        for(int j=M-1;j>=0;j--) {
                                if(pat.charAt(j)!=txt.charAt(i+j)) {
                                        skip=Math.max(1, j-right[txt.charAt(i+j)]);
                                        break;
                                }//if
                        }//inner for
                        if(skip==0) return i;//found
                }//Outer for
                return N;//not found
        }//search

        public static void main(String[] args) {
                String txt="ABAAABCD";
                String pat="ABC";
                BoyerMooreDemo bm=new BoyerMooreDemo(pat);
                int offset=bm.search(txt);
                System.out.println("Pattern found at index : "+offset);


        }

}
```