No.

Date

KMP算法（模式匹配）

1、目的：字符串匹配（模式匹配）中，实现对原字符串t的无回溯匹配。

2、算法：对模式P，当 $p_i \neq t_j$ 时，直接右移 $i-next[i]$ 个字符，如果 $next[i]$ 大于等于0，则从 t_j 与 $p_{next[i]}$ 继续比较，否则从 t_{j+1} 与 p_0 比较下去。

3、内核：

```
int pMatch(t, p, next[]);
{ while(i<p->n && j<t->n)
    if(i==-1 || p->c[i]==t->c[j])
      {i++;j++;}
    else i=next[i];
  if(i>=p->n) return (j-p->n+1);
  else return 0;
}
```

匹配模式在t中的序号（第一个字符从1开始）

4、算法：计算next数组：next[i]的值是 $p_0 \cdots p_{i-1}$ 的最大相同前后缀的长度。

① 对任何p，next[0]=-1；

② next[i]<i；（不可能向右移）

③ 若next[i]=k，则 next[i+1]≤k+1；

内核：

```
void makeNext(p, next[])
{ k=-1, next[0]=-1;
  while(i<p->n-1){                    (计算next[i+1])
    while(k>=0 && p->c[i]!=p->c[k]) k=next[k];   (继续缩进)
    i++; k++;
    if(p->c[i]==p->c[k]) next[i]=next[k];
    else next[i]=k;
  }
}
```

（若 $P_{next[i]}=P_i$，则应该再移，令 k=next[k]）

5、图示① $p=$"abcaababc"，$p\to n=$ [redacted]9

下标 i	0	1	2	3	4	5	6	7	8	改进后的next数组中i的值已不再表示$P_0 \sim P_{i-1}$中最大相同前后缀的长度了，这在$P_i=P_{next[i]}$时体现出来。
P_i	a	b	c	a	a	b	a	b	c	
$P_0 \sim P_{i-1}$的k值				0		1		1	2	
最后的判断→ P_k与P_i		≠	≠	=	≠	=	≠	=	=	
next[i]	-1	0	0	-1	1	0	2	0	0	

$i=0$ $k=-1$

→ $i=1$ $k=0$ next[1] = 0.

→ $P_i \neq P_k$, $k=-1$

→ $i=2$ $k=0$ next[2] = 0

→ $P_i \neq P_k$, $k=-1$

→ $i=3$ $k=0$ $P_i=P_k$, next[3] = next[0] = -1

→ $i=4$ $k=1$ $P_i \neq P_k$, next[4] = 1

→ [redacted] $P_i \neq P_k$, $k=0$

→ $P_i=P_k$

→ $i=5$ $k=1$ $P_i=P_k$, next[5] = next[1] = 0.

→ $P_i=P_k$ [redacted]

→ $i=6$ $k=2$ [redacted]

→ $P_i \neq P_k$, next[6] = 2

→ $P_i \neq P_k$, $k=$ next[2] = 0.

→ $P_i=P_k$.

→ $i=7$ $k=1$ $P_i=P_k$, next[7] = next[1] = 0.

→ $i=8$ $k=2$ $P_i=P_k$, next[8] = next[2] = 0

② t = "aabcbabcaabcaababc" t->n=18 ,p同前例.

i=	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
t=	a	a	b	c	b	a	b	c	a	a	b	c	a	a	b	a	b	c
p	a	b	c	a	a	b	a	b	c									
	✓	×																
next[1]=0		a	b	c	a	a	b	a	b	c								
		✓	✓	✓	×													
next[3]=-1						a	b	c	a	a	b	a	b	c				
						✓	✓	✓	✓	✓	✓	×						
next[6]=2										a	b	c	a	a	b	a	b	c
												✓	✓	✓	✓	✓	✓	✓

"i" ≥ p->n , return j-p->n+1 = 18-9+1=10 ;

时 O(m+n) 空 O(m)