

串

KMP算法：构造next[]表

13-C4

一切都是暂时的，转瞬即逝  
而那逝去的将变为可爱

邓俊辉

deng@tsinghua.edu.cn

# 減而治之

-1 0 0 1 2 3 1 0  
M A M A M M I A

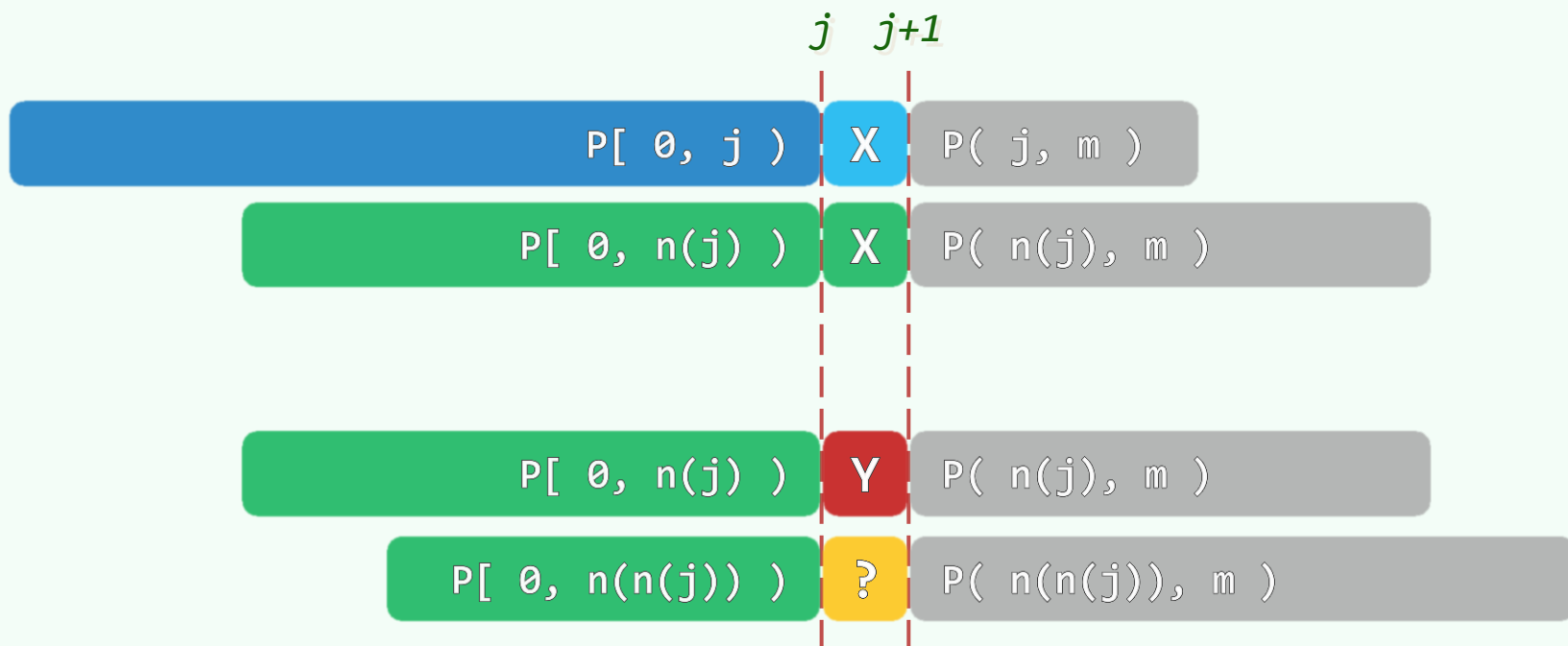
-1 0 0 1  
M A M A M M I A  
M A M A M M I A

-1 0 0 1 2  
M A M A M M I A  
M A M A M M I A

-1 0 0 1 2 3  
M A M A M M I A  
M A M A M M I A

-1 0 0 1 2 3 1  
M A M A M M I A  
M A M A M M I A  
M A M A M M I A  
M A M A M M I A

$$next[j + 1] = next[j] + 1 \quad \text{iff} \quad P[j] = P[next[j]]$$



# 算法实现

```
❖ int * buildNext( char * P ) {
    size_t m = strlen(P), j = 0;
    int *N = new int[m];
    int t = N[0] = -1;
    while ( j < m - 1 )
        ( 0 > t || P[j] == P[t] ) ?
            N[ ++j ] = ++t : t = N[t];
    return N;
}
```

$j$   $j+1$

\*

$P(-1, m) = P[0, m)$

$P[0, n(n(n(j))) )$

X

$P(n(n(n(j))), m)$

$P[0, n(n(j)) )$

Z

$P(n(n(j)), m)$

$P[0, n(j) )$

Y

$P(n(j), m)$

$P[0, j )$

X

$P(j, m)$