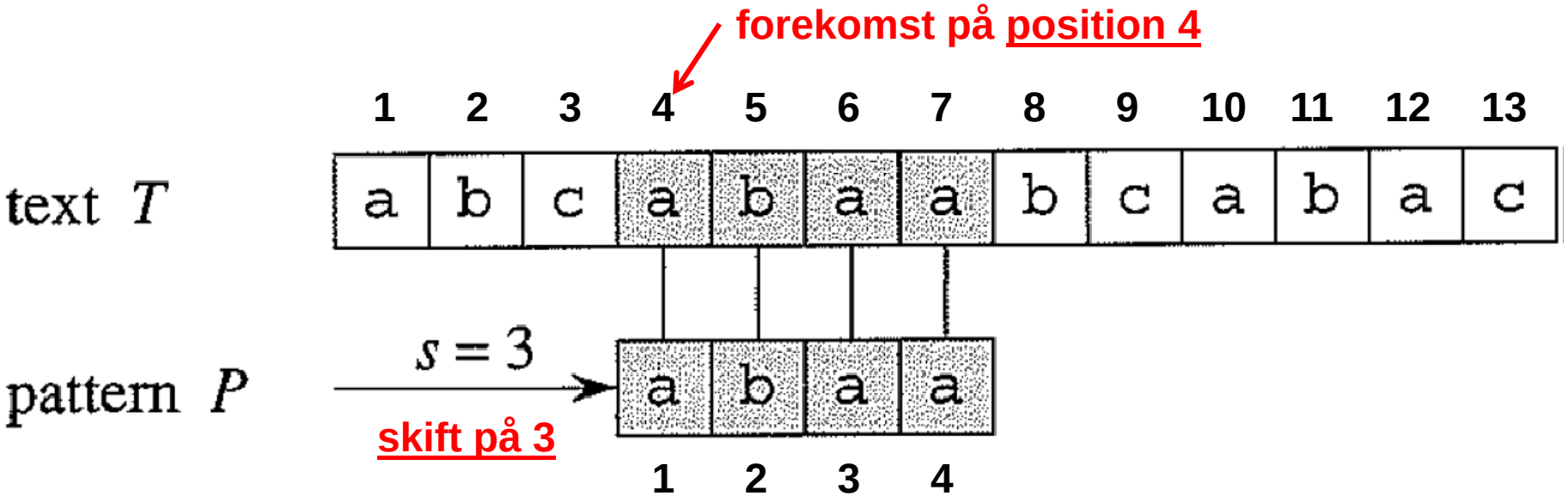
Algoritmer og Datastrukturer 2

Mønstergenkendelse [CLRS, kapitel 32.1-32.2, 32.4]

Gerth Stølting Brodal

Aarhus Universitet

Mønster genkendelse



Input: Tekst T af længde n og mønster P af længde m

Output: Alle positioner i T hvor P forekommer

Naive Algoritme

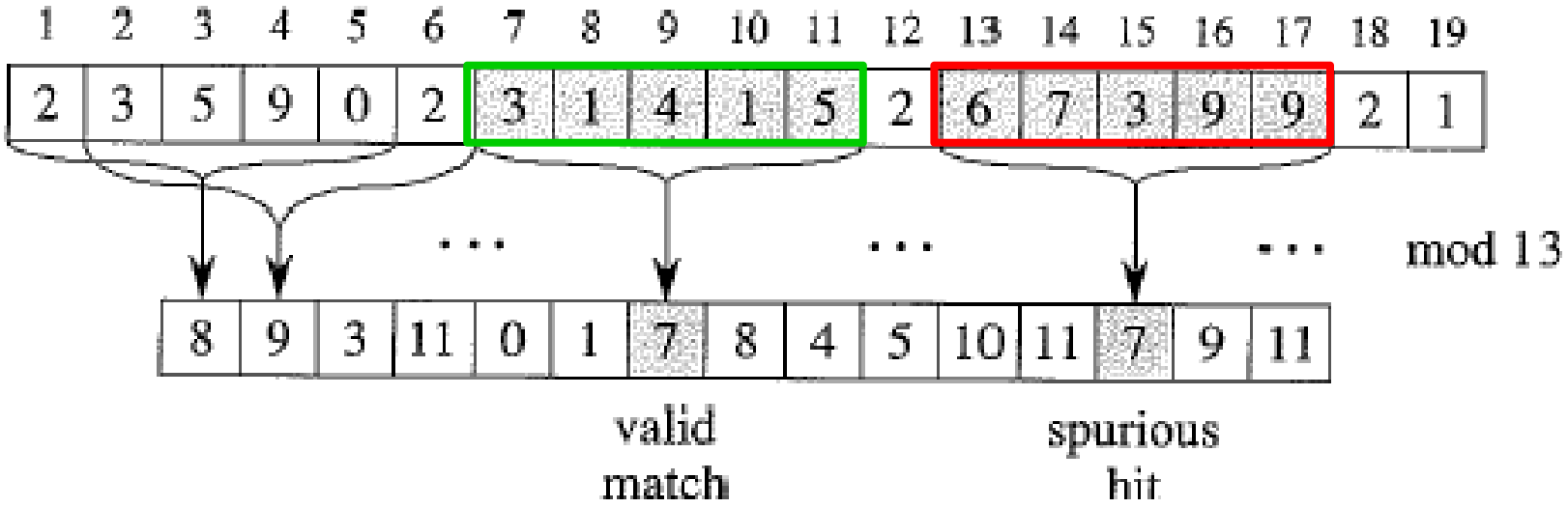
NAIVE-STRING-MATCHER(T, P)

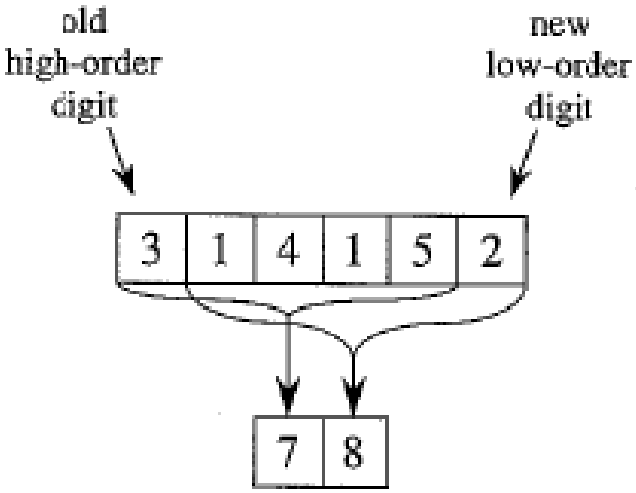
```
1  n ← length[T]
2  m ← length[P]
3  for s ← 0 to n − m
4        do if P[1..m] = T[s + 1..s + m]
5              then print "Pattern occurs with shift"
```

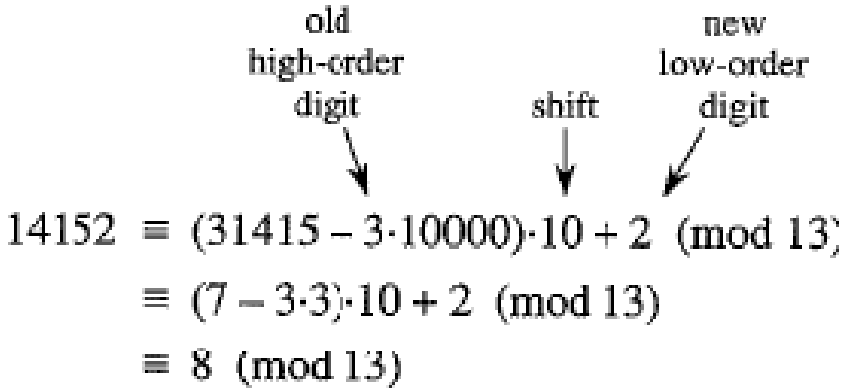
Rabin-Karp

```
RABIN-KARP-MATCHER(T, P, d, q)
 1  n ← length[T]
 2  m ← length[P]
 3  h ← d^(m−1) mod q
 4  p ← 0
 5  t_0 ← 0
 6  for i ← 1 to m                          ▷ Preprocessing.
 7       do p ← (dp + P[i]) mod q
 8          t_0 ← (dt_0 + T[i]) mod q
 9  for s ← 0 to n − m                      ▷ Matching.
10       do if p = t_s
11             then if P[1..m] = T[s + 1..s + m]
12                       then print "Pattern occurs with shift" s
13          if s < n − m
14             then t_{s+1} ← (d(t_s − T[s + 1]h) + T[s + m + 1]) mod q
```

Rabin-Karp: Eksempel



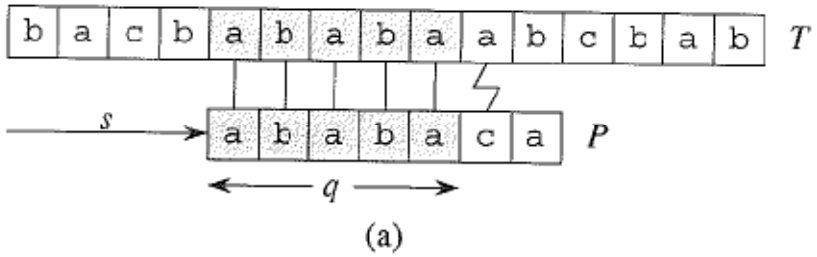




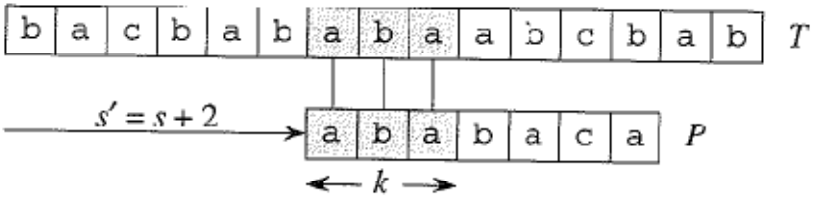
Knuth-Morris-Pratt

```
KMP-MATCHER(T, P)
 1  n ← length[T]
 2  m ← length[P]
 3  π ← COMPUTE-PREFIX-FUNCTION(P)
 4  q ← 0                                  ▷ Number of characters matched.
 5  for i ← 1 to n                         ▷ Scan the text from left to right.
 6       do while q > 0 and P[q + 1] ≠ T[i]
 7            do q ← π[q]                  ▷ Next character does not match.
 8          if P[q + 1] = T[i]
 9            then q ← q + 1               ▷ Next character matches.
10          if q = m                       ▷ Is all of P matched?
11            then print "Pattern occurs with shift" i − m
12                 q ← π[q]                ▷ Look for the next match.
```

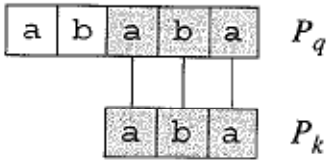
Knuth-Morris-Pratt: Eksempel



(a)



(b)



(c)

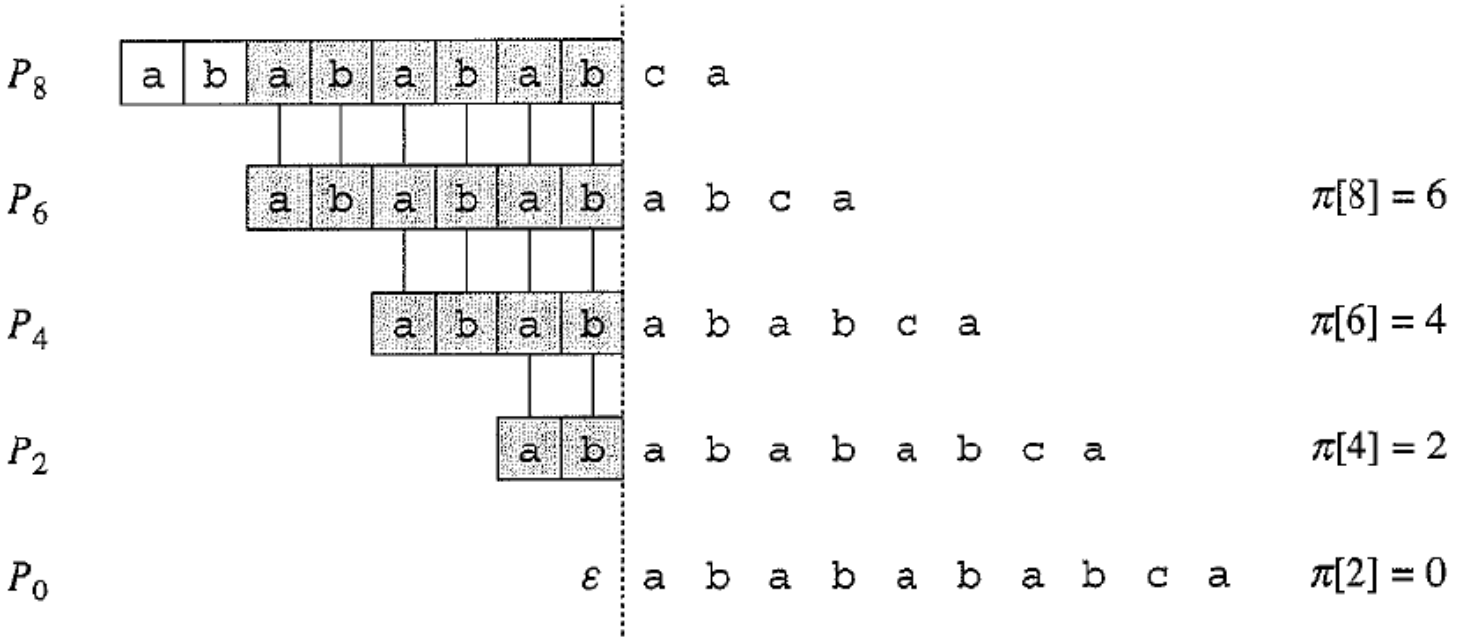
Knuth-Morris-Pratt: Beregning af prefix funktionen

```
COMPUTE-PREFIX-FUNCTION(P)
 1  m ← length[P]
 2  π[1] ← 0
 3  k ← 0
 4  for q ← 2 to m
 5        do while k > 0 and P[k + 1] ≠ P[q]
 6              do k ← π[k]
 7           if P[k + 1] = P[q]
 8              then k ← k + 1
 9           π[q] ← k
10  return π
```

Knuth-Morris-Pratt: Beregning af prefix funktionen

i	1	2	3	4	5	6	7	8	9	10
$P[i]$	a	b	a	b	a	b	a	b	c	a
$\pi[i]$	0	0	1	2	3	4	5	6	0	1

(a)



(b)

Worst-case tider

Algorithm	Preprocessing time	Matching time	[CLRS]
Naive	0	$O((n-m+1)m)$	32.1
Rabin-Karp	$\Theta(m)$	$O((n-m+1)m)$	32.2
Finite automaton	$O(m \lvert\Sigma\rvert)$	$\Theta(n)$	(32.3)
Knuth-Morris-Pratt	$\Theta(m)$	$\Theta(n)$	32.4