Substring Search:-

Boyer Moore:-

1) Bad symbol shift table (d_1)
last is ignored, otherwise character repeated ignore.

(i) BAB

c	B	A	D
t(c)	2	1	3

GOOD

c	G	O	D
t(c)	3	1	4

(ii)

c	D	I	S	G	U	S	T	I	N	G
t(c)	9			6	5	4	3	2	1	10

G is repeated but last is ignored so the G-6 is taken.

// for all other remaining characters in text the safe no. of shifts is size of pattern

	0	1	2	3	4	5	6	7	8	9
* C	C	O	N	S	I	S	T	I	N	G
t(c)	9	8			4	3	2	1	10	

for all other characters in the text the shift is 10

② Good suffix shift table.

k	pattern	d_2
1	A B C B A B	2.
2	A B C B A B	4
3	A B C B A B	4
4	A B C B A B	4
5	A B C B A B	4

A B

AB AB

ABC BAB.

k	pattern	d_2
1	10000	3
2	10000	2
3	10000	1
4	10000	5

k	pattern	d_2
1	00001	5
2	00001	5
3	00001	5
4	00001	5

1	0
10	00
100	000

0	1
00	01
000	001

(i) 01010.

(ii) TORONTO.

d_1 $K=0$.

$d_1 = \max\{T(c) - K, 1\}$.

otherwise

$\max(d_1, d_2)$

k from 0 – one less because all matched otherwise

shift $d = \begin{cases} d_1 & k = 0 \\ \max\{d_1, d_2\} & \text{otherwise} \end{cases}$

$d_1 = \max\{T(c) - k, 1\}$

* Apply Boyer Moore algorithm for.

TEXT:-

BESS_KNEW_ABOUT_BAOBABS.

PATTERN:

BAOBAB.

d_1

c	A	B	c	D	---	o	. .	z	-
t(c)	1	2	6	6		3		6	6

k	pattern	d_2.		
0+1	BAOBAB	2		
1+1	BAOBAB	5 (B)	B	B.
2+1	BAOBAB	5 (B)	BA	AB.
3+1	BAOBAB	5	BAO	BAB
4+1	BAOBAB.	5	BAOB	OBAB
5	~~BAOBAB~~			

text: B E S S _ K N E W _ A B O U T _ B A O B A B S

pat: B A O B A B

(i) $B \neq K$.

$d_1 = \{\max\{6-0, 1\}\}$

$K = 6$ 0 - no. of matched characters.

soshift by 6.

(ii) $B = B$ (11)

$A = A$ (10)

2 - matched characters $= K = 2$

$d_1 = \max\{6-2, 1\} = 4$

$d_2 = K = 2 = 5$

$= \max(d_1, d_2)$

Shift by 5

(ii) $B = B$ (16). $K = 1$ matched characters

$d_1 = 6 - 1 = (5, 1)$ max = 5

$d_2 = 2$ $(K = 1)$

max (5, 2)

Shift by 5.

(iii) from 16 to 21

matches return 16

No. of character shift - first table.
No. of prefix and suffix - second table.

Only d_1 - horspool algorithm (shift based on d_1).

* Knuth - Morris - Pratt algorithm. (KMP)

Find the length of the longest proper prefix in the sub pattern that matches a proper fix in the same sub-pattern

π- table or prefix table

char	A	B	A	B
index	0	1	2	3
value	0	0	1	2.

* left to right
(not right to left)
left to right substring

substring	proper prefix	proper suffix	length / Π-value
A	NULL	NULL	0
AB	A	B	0
ABA	A, AB	A, BA	1
ABAB	A, AB, ABA	B, AB, BAB	2

The shift in KMP algorithm is computed by the formula

$$k - \Pi[k-1]$$

where k is the no. of matched character.

* For a given text search for the pattern
abababca

T: b a c b a b a b a b a c a a b

char	a	b	a	b	a	c	a
index	0	1	2	3	4	5	6
value	0	0	1	2	3	0	1

	substring	PP	PS	Π. value
1	a	NULL	NULL	0.
2	ab	a	b	0.
3	aba	a, ab	ba, a	1
4	abab	a, ab, aba	b, bab, ab	2
5	ababa	abab	ba, baba, a, aba	3.
6	ababac			0
7	abababa			1

6. PP

a, ab, aba, abab, ababa

PS.

c, ac, bac, abac, babac

7 PP

a, ab, aba, abab, ababa, ababac

SS.

a, ca, aca, baca, abaca, babaca

0 1 2 3 4 5 6

i) b a c b a b a b a b a c a a b.

*

a b a b a c a

first character mismatch. (K=0)

so shift by 1

ii) a = a (1)

b ≠ (c)

K = 1

Shift = 1 − π[K−1] = 1 − π[0] = 1

iii) a ≠ c.

shift by 1.

iv) a ≠ b.

shift by 1.

v) a = a, b = a, a = a, b = b, a = a,

b ≠ c

K = 5

shift = 5 − π[5−1] = 5 − π[4] = 5 − 3 = 2

↗ if multiple occurence then slift by 1

v) Everything match at index 6.

Rabin-Karp Algorithm:

text: 1 5 9 6 7 9 26 4 3 ---

pattern 26

prime 11

hash(pattern) = 4. = 26 mod 11 = 4.

26%.11 = 4

* shift = 0

1 5 9 6 7 9 26 43 - - -

11

26.

hash(15) = 4 (hit) (spurious hit)
No match.

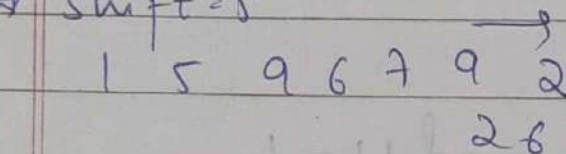
* shift (1)
hash(59) = 59%.11 = 4 (spurious hit)
No match

* shift (2)
hash(96) = 8 (No compare)

* shift = 3
hash (67) = 1
No compare.

* shift = 4
hash(79) = 2 No compare

* shift = 5



hash(92) = 4 (spurious hit)

* shift = 6
hash(26) = 4
Compare 26
Match found.

Ex * y = 25 z = 26
a = 1 b = 2 c = 3
d = 4 e = 5 f = 6
g = 7 h = 8 i = 9
j = 10 K = 11 l = 12
m = 13 n = 14 o = 15
p = 16 q = 17 r = 18
s = 19 t = 20 u = 21
v = 22 w = 23 x = 24

O(m*n) - Worst case for all.

pattern doesn't exist - best - O(n+m) - Boyer Moore.

Repeating - KMP