

Explaining hash function:-

$\text{hashValue} = (\text{hashValue} + (\text{char} \times t^i)) \% q$  for each character in the string, updating  $t^i = (t^i \times t) \% q$ .

Hash function explanation-

Step 1) - Initialisation

- Start with 'hashValue = 0'.
- Set 'tpower = 1' (this represents  $t^1$ , the current of 't')

Step 2) - Iterate through each character

- For each character in the string, update the hash value using:-

$$\text{hashValue} = (\text{hashValue} + (\text{char} \times t^i)) \% q$$

- Here, 'char' is the ASCII value of the character  $t^i$  is the current power of 't' and 'q' is a large prime number to keep hash values within a manageable range.

Step 3) - Update the power of 't'.

- After processing each char, update  $t^i$  to  $t^{i+1}$ :  
 $t^i = (t^i \times t) \% q$

eg,

- Text : 'AABAB'
- Pattern : 'AB'
- Parameters:
  - $t = 31$
  - $q = 1000000007$

Rolling hash for 'AABAB'

- 1). Initialise  $tpower = 1$  and  $hashvalue = 0$ .
- 2). Calculate the hash value for 1<sup>st</sup> window 'AB':
  - 'A' (ASCII 65):
  - $hashvalue = (0 + (65 \times 1) \cdot 1000000007) \cdot 1000000007 = 2111$
  - $tpower = (31 \times 31) \cdot 1000000007 = 961$

The hash value for window 'AB' is 2111.

3). Now, roll the hash over text -

- Remove contribution of the outgoing char and add the incoming char's contribution for each subsequent window.

### Second window "BA"

- Remove 'A' (ASCII 65) -

$$\text{hashValue} = (2111 - 65 \times 961) \% (1000000007 + 1000000007) \\ = 7.1000000007$$

- Outgoing contribution =  $(65 \times 961) \% 1000000007 = 62465$

$$\text{hashValue} = (2111 - 62465 + 1000000007) \% 1000000007 \\ = 9376543653$$

- Add 'A' (ASCII 65) -

$$\text{hashValue} = (9376543653 \times 31 + 65) \% 1000000007 \\ = 650000005$$

Hash for "BA" is 650000005 (no match).

Third window "AB",

- Remove 'A' (ASCII 65)

- Add 'B' (ASCII 66)

$$\text{hashValue} = (64936579 \times 31 + 66) \% 1000000007 \\ = 2013010$$

Hash for "AB" is 2111 (match found).