Submitted by AMAN SONI

RA1811003010363

TP202 CSE-F1

(2) Input string ~~AB~~ADBDA

p = AB m = 2

q = 11 Initially h, t = 0.

d = 5

n = 5^(2-1) % 11 = 5

Finding the value of p for the pattern to be searched B.

Iteration 1:

P = (d x initial value of h) + ((ASCII of char)) % q

P = (~~5x0~~ 66) % 11 : ASCII value of B = 66

= 0

Iteration 2: P = (5X0 + 68) % 11 (ASCII value of D = 68)

= 2

hash code for [BD] = 2

for searching value in input string ~~BA~~ (i.e. AD) = 8

i.e. not equal to p (t ≠ p)

∴ not found.

Iteration 3: for the next char, the process will be discarding the previous character and including the next char.

t = d x (prev(t) - (ASCII of prev. char) * h) **

((ASCII of Next char)) % q

t = (5(8 - 65 x 5) + 66) % 11

t = -1 (Add q to it)

So t = -1 + 11 = 10

So t ≠ p. (next iteration)

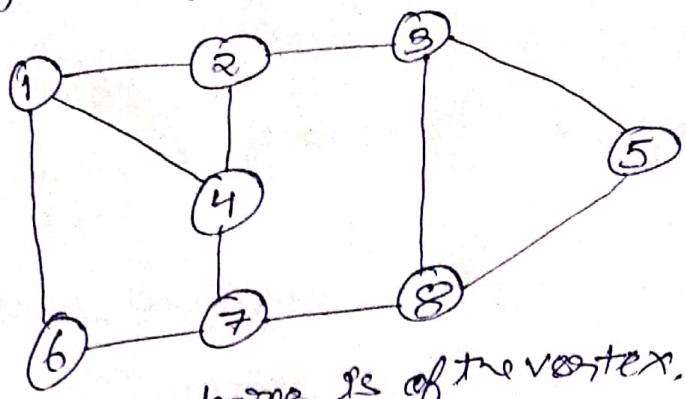
Iteration 4: $t = (5 \times 11 - 63 \times 5 + 68) \% 11$

$t = -9$ $\quad t = -9 + 11 = 2$

So $t = p$ we can say that the element is present in the input string.

String was present at index value 2.

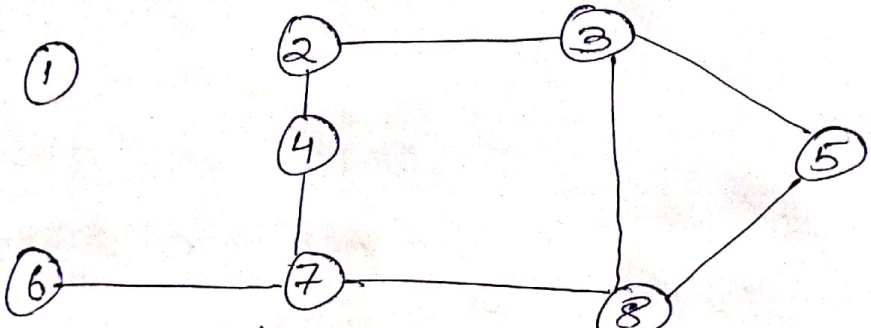
1) Minimum vertex cover of following undirected graph by greedy approach.



Edge	Degree
1	3
2	3
3	3
4	3
5	2
6	2
7	3
8	3.

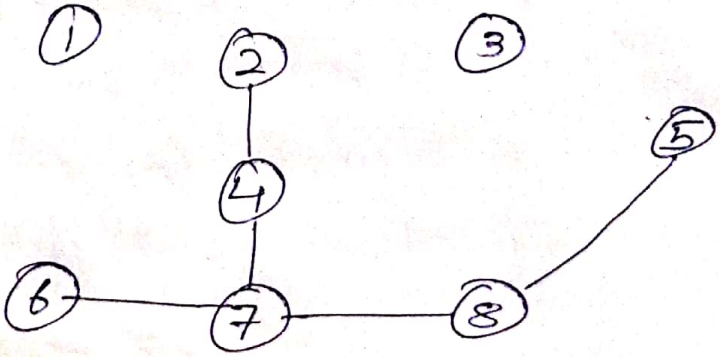
⇒ The max degree is of the vertex. 1, 2, 3, 4, 7 and 8. Here, I am chosing (1).

⇒ Now adding vertex to the solution and remove the ~~solution~~ vertex along with its incident edges from the graph.



Edge	Degree
1	0
2	2
3	3
4	2
5	2
6	1
7	3
8	3

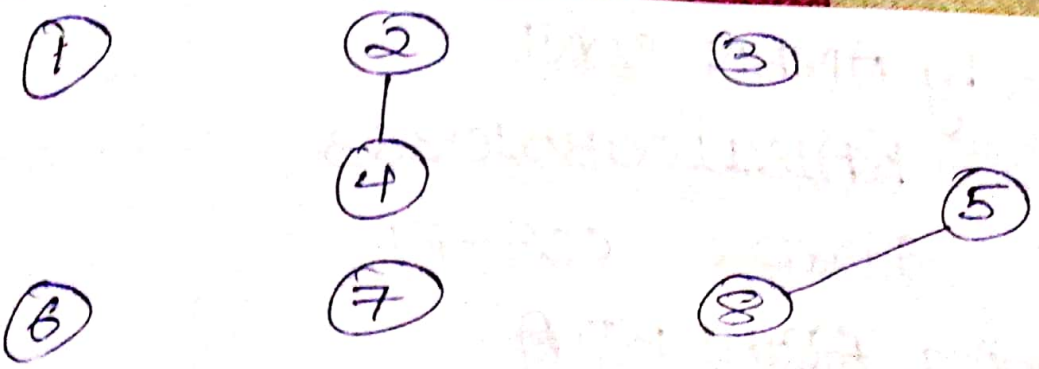
⇒ Now, the max degree is of vertex 3, 7 and 8. I am chosing (3).



Edge	Degree
1	0
2	1
3	0
4	2
5	1
6	1
7	3
8	2

⇒ Here, the max degree is of the vertex 7. So chosing (7).

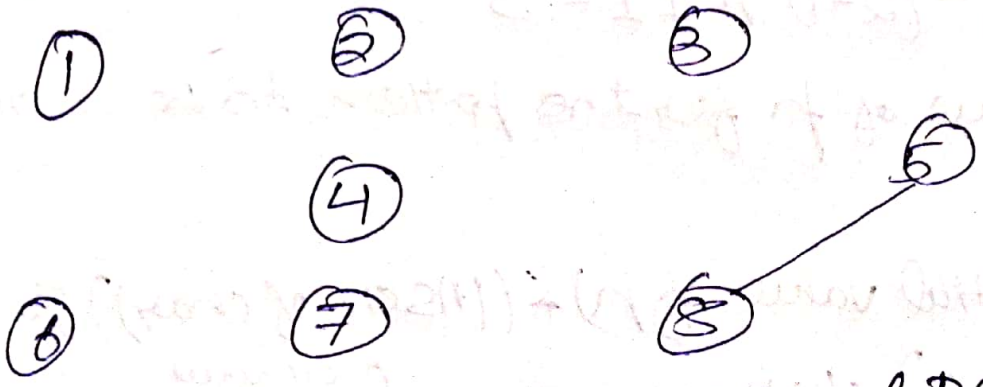
The resultant will be :



Edge	Degree
1	0
2	1
3	0
4	1
5	1
6	0
7	0
8	1

Here the max degree is of the vertex 2, 4, 5 and 8. So I am chosing (2).

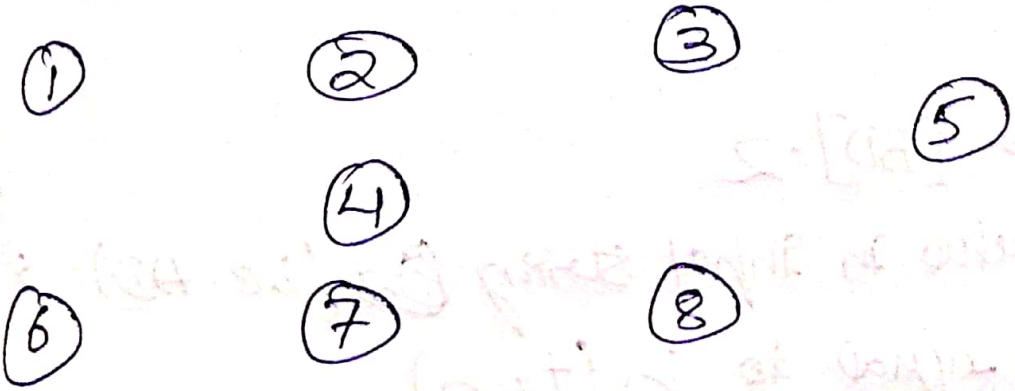
Therefore:



Edge	Degree
1	0
2	0
3	0
4	0
5	1
6	0
7	0
8	1

Here the maximum degree is of the vertex 5 and 8. So I am chosing (5).

The resultant is.



The vertex cover is $\{1, 3, 7, 2, 5\}$.