Date _____ Page _____

Day - 214

Graph - 18

* Circle of String:

1 → ab bc cd da

⇒ We have make a circle by chaining the strings.

⇒ But the condition is last char of the string is same as first char of another string then the chain will be formed.

⇒ So,

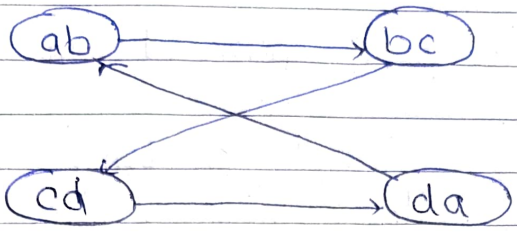
ab – bc – cd – da

⇒ 1

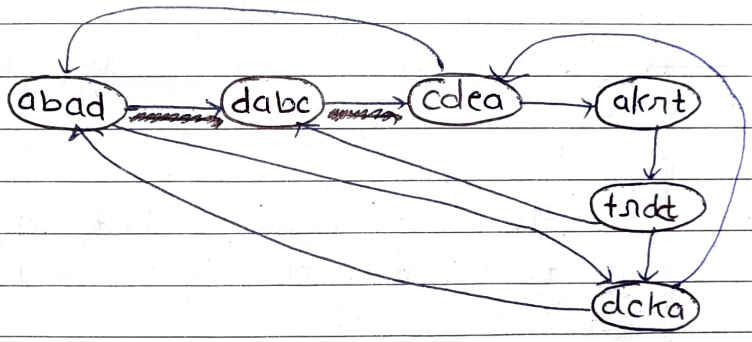
⇒ It is not necessary that only first string is used for starting, any string can be used.

⇒ So, first we make nodes of strings.

Date _____ Page _____



⇒ Ex: abad dabc cdea akrt trdd dcka



⇒ Making edges helps in finding all the possible paths.

⇒ So, this is called Hamiltonian cycle in which we visit every node exactly once and return to the source node.

⇒ So, we visit every path and check for the condition.

⇒ This will take n! time.

⇒ Reason for the high time complexity is that we finding & going on every path.

⇒ So, the chances of repeating a particular path is very high.

Date _____ Page _____

⇒ So, now for optimized answer —

⇒ We can see that we don't have require chars other first & last.

⇒ So, our new string will be —

⇒ ad dc ca at td da

⟶ a d c a t d

First char

⇒ So, the last char is just saying that go to that string of name of the last char.

⇒ So, we can visualize it as —

a $\xrightarrow{abad}$ d

d $\xrightarrow{dabc}$ c

c $\xrightarrow{cdeq}$ a

a $\xrightarrow{akrt}$ t

t $\xrightarrow{trdd}$ d

d $\xrightarrow{dcka}$ a

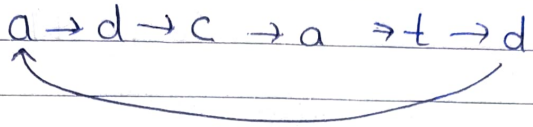
⇒ So, we are doing this —

abad

⇒ a → d

Date _____ Page _____

⇒ Then,



⇒ Now, find eulerian circuit & check if the start == end.

⇒ For directed graph, we check indegree & outdegree for euler circuit.

⇒ And then do ~~BFS~~ DFS operation.

⇒ Also, we represent 'a' as 0, 'b' as 1, and so on.