

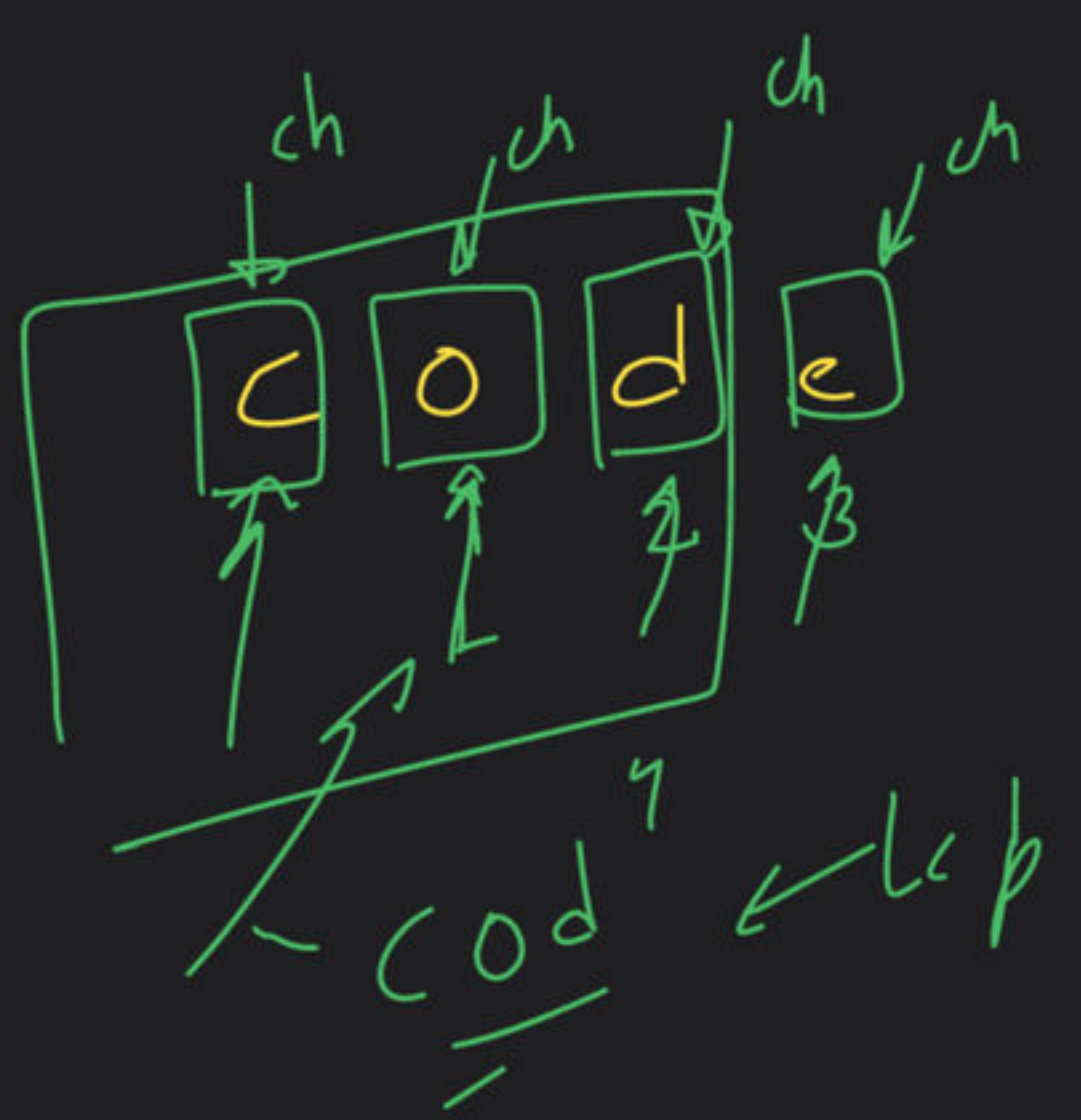


Hashmaps & Tries - Class 3

Special class

→ Trie → D.S → Multiway tree → Pattern matching

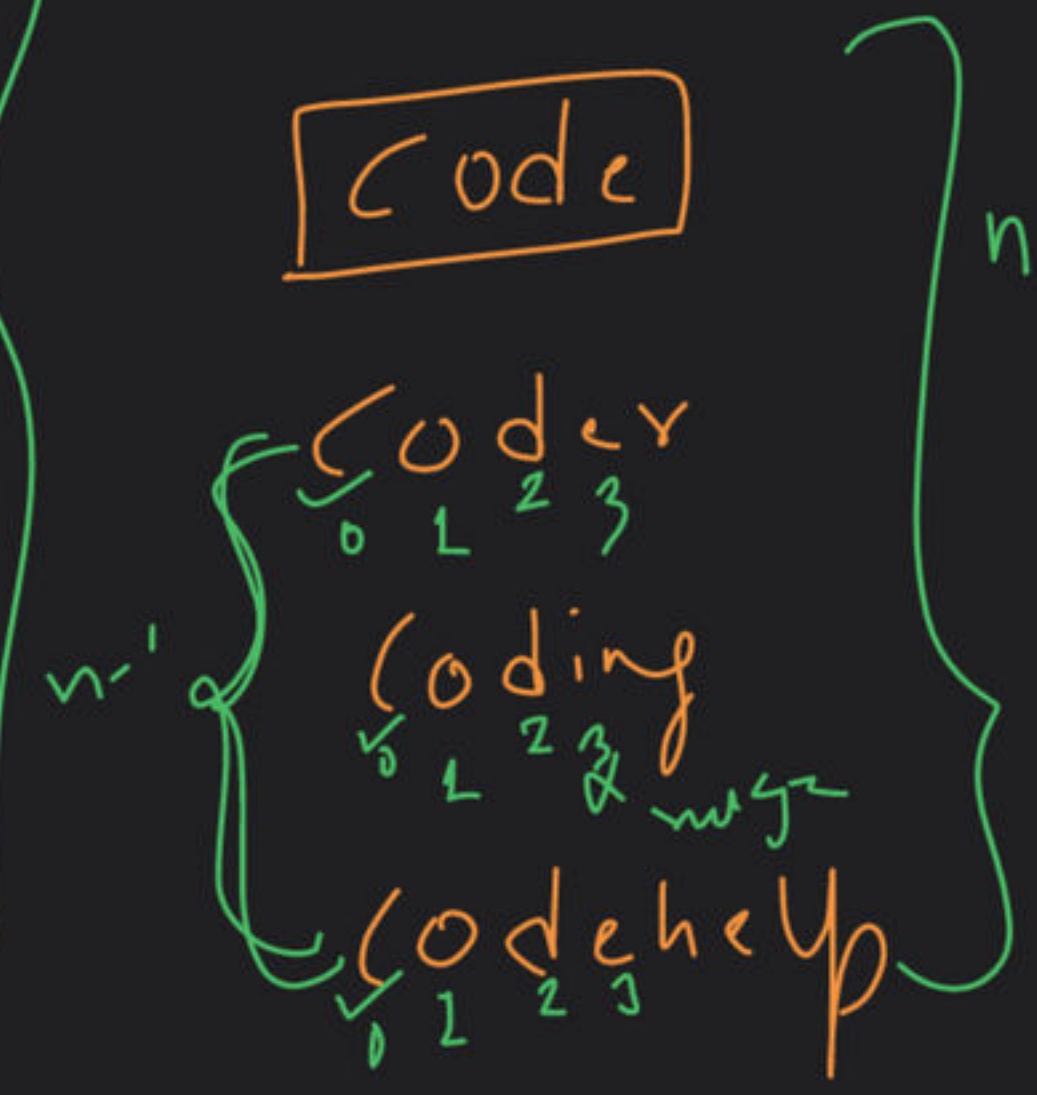
#1

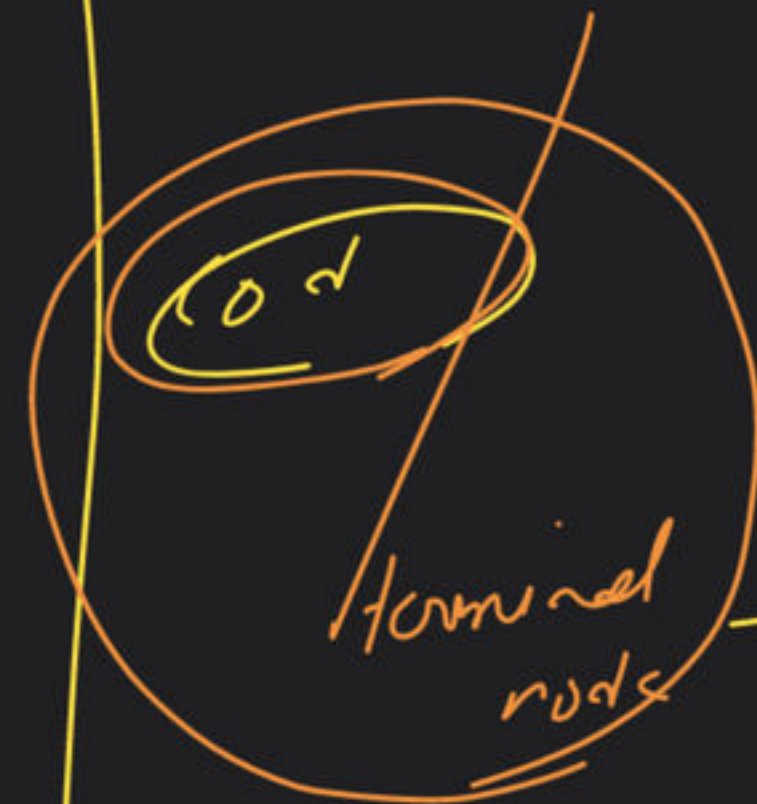
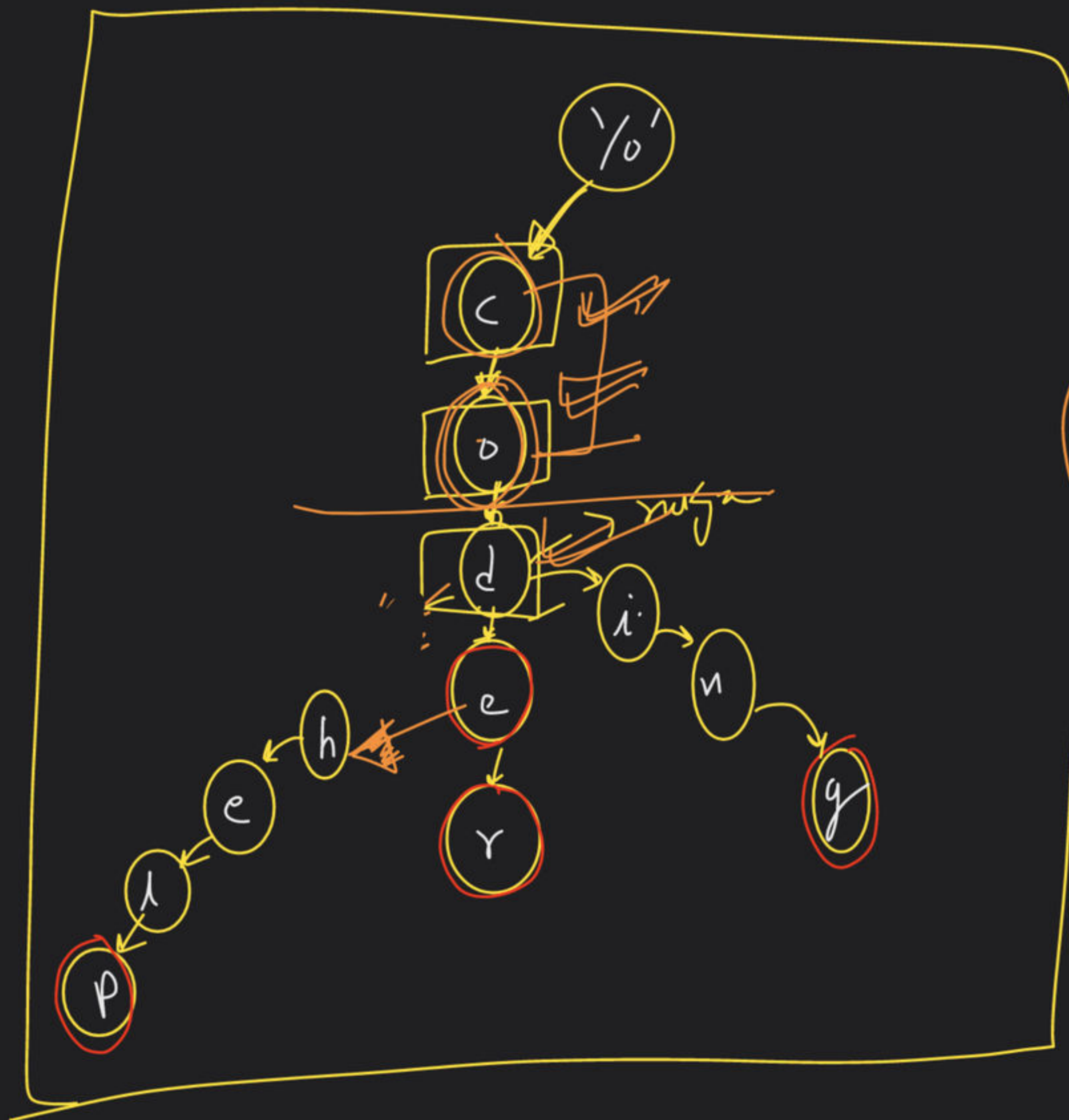


```

for (
{
    ch →
    for (n-1, try)
    {
        // match
    }
}

```





LCP

code ✓
coder ✓
coding ✓
codchelp ✓

CO

3000

CO



Love

Love songs ✓

Love Babbar ✓

Love quotes ✓

Loving ✓

Lovely ✓

suggestion

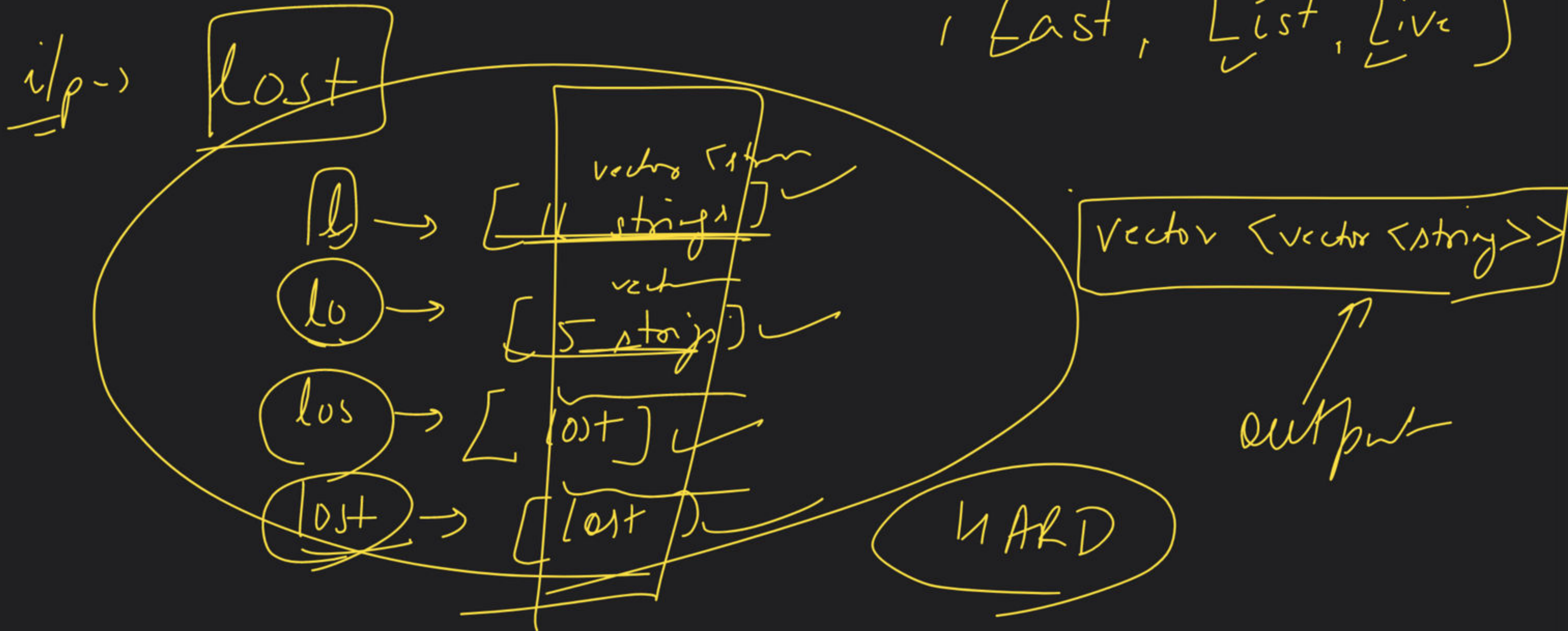
val[] \rightarrow { ~~coding~~, coder, codendp, code, codeverse,
codingDunic,
codeforces }

i/p \rightarrow code f

<u>code f</u>	\rightarrow	[7 strings]
<u>code</u>	\rightarrow	[7 str]]
<u>cod</u>	\rightarrow	[7 str]]
<u>code</u>	\rightarrow	[5 str]]
<u>code f</u>	\rightarrow	[codeforces]]

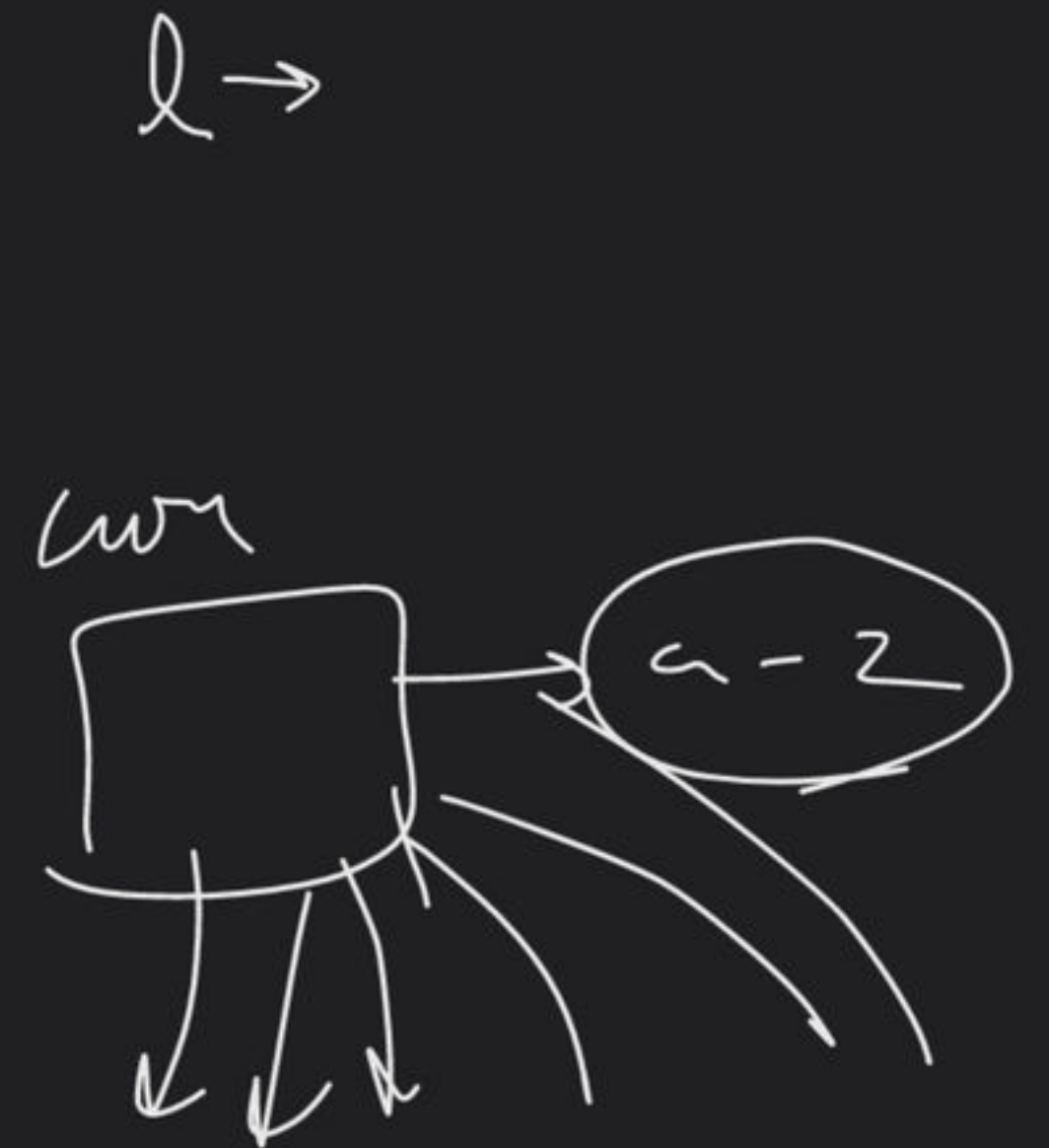
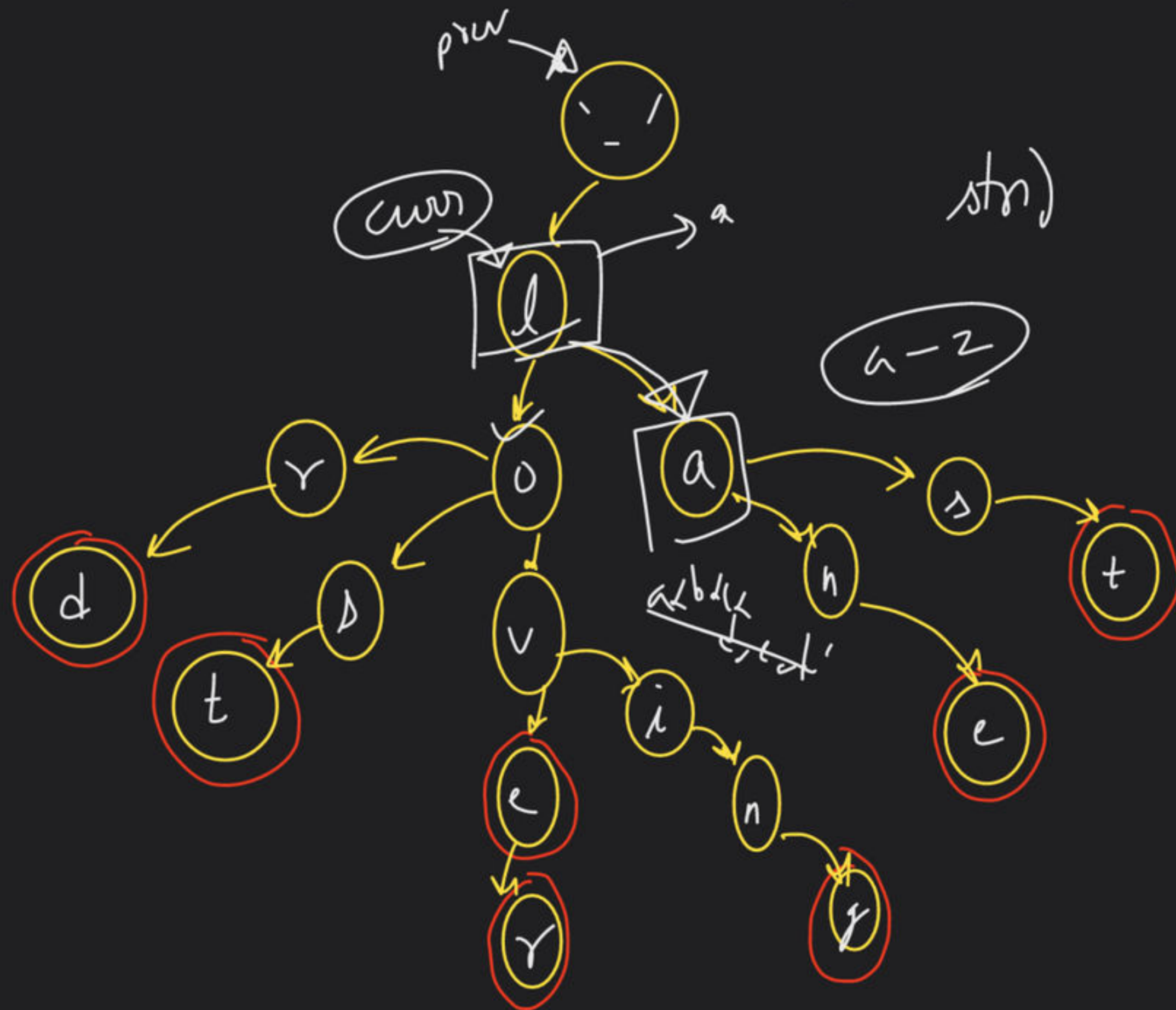
Vector → [lover, Lanc, Lost, Lend, Loving]

Love, Lord, Least
[Last, List, Live]

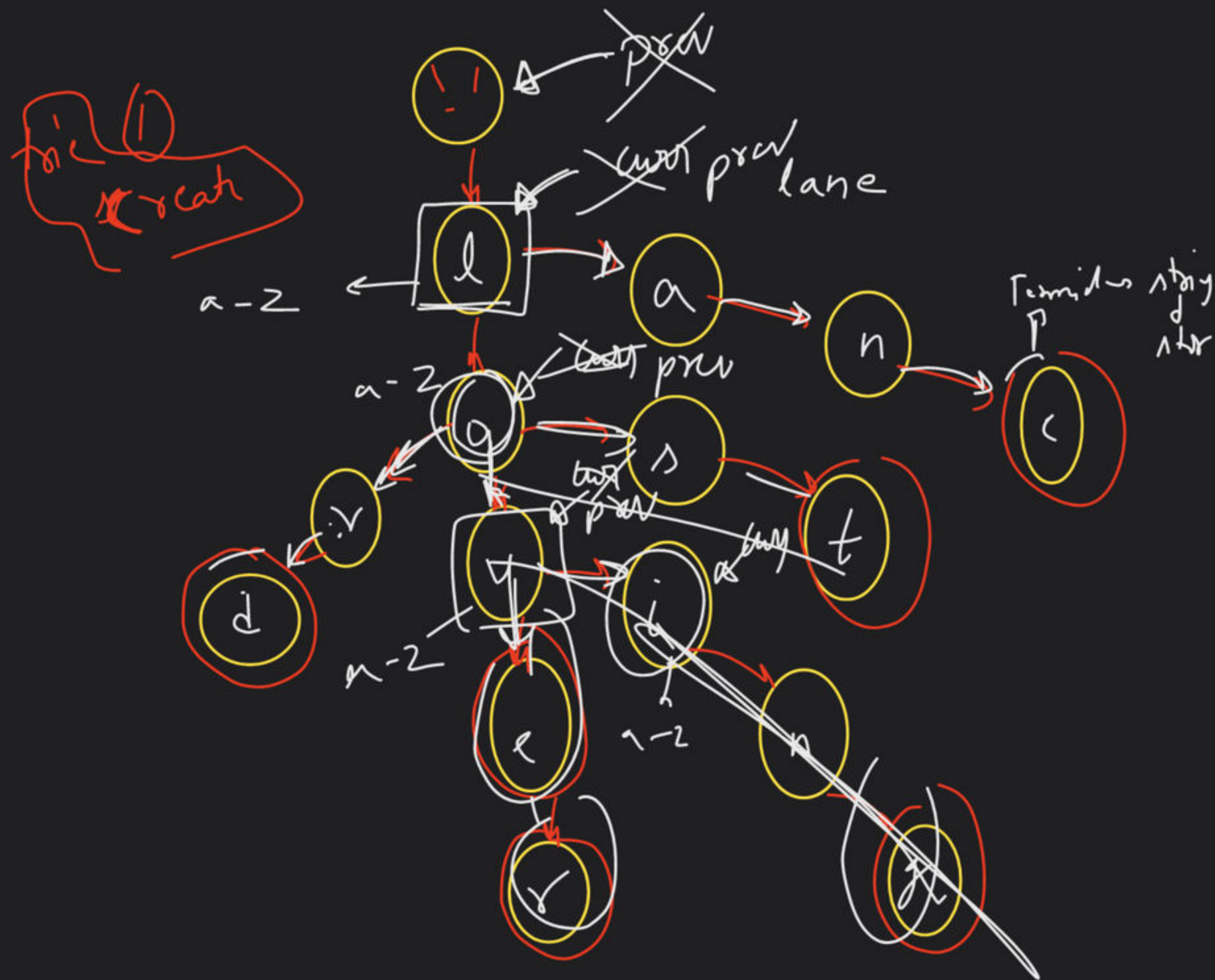


Vectors \rightarrow [love, lover, loving, lane, last, lost, lord]

i/p \rightarrow "lovi"



vec \rightarrow [Love, lover, Loving, lost, ~~last~~, lane, lord]



lovi'

$l \rightarrow [low, low, low, low, low, low, low, low]$
 $low = 1$

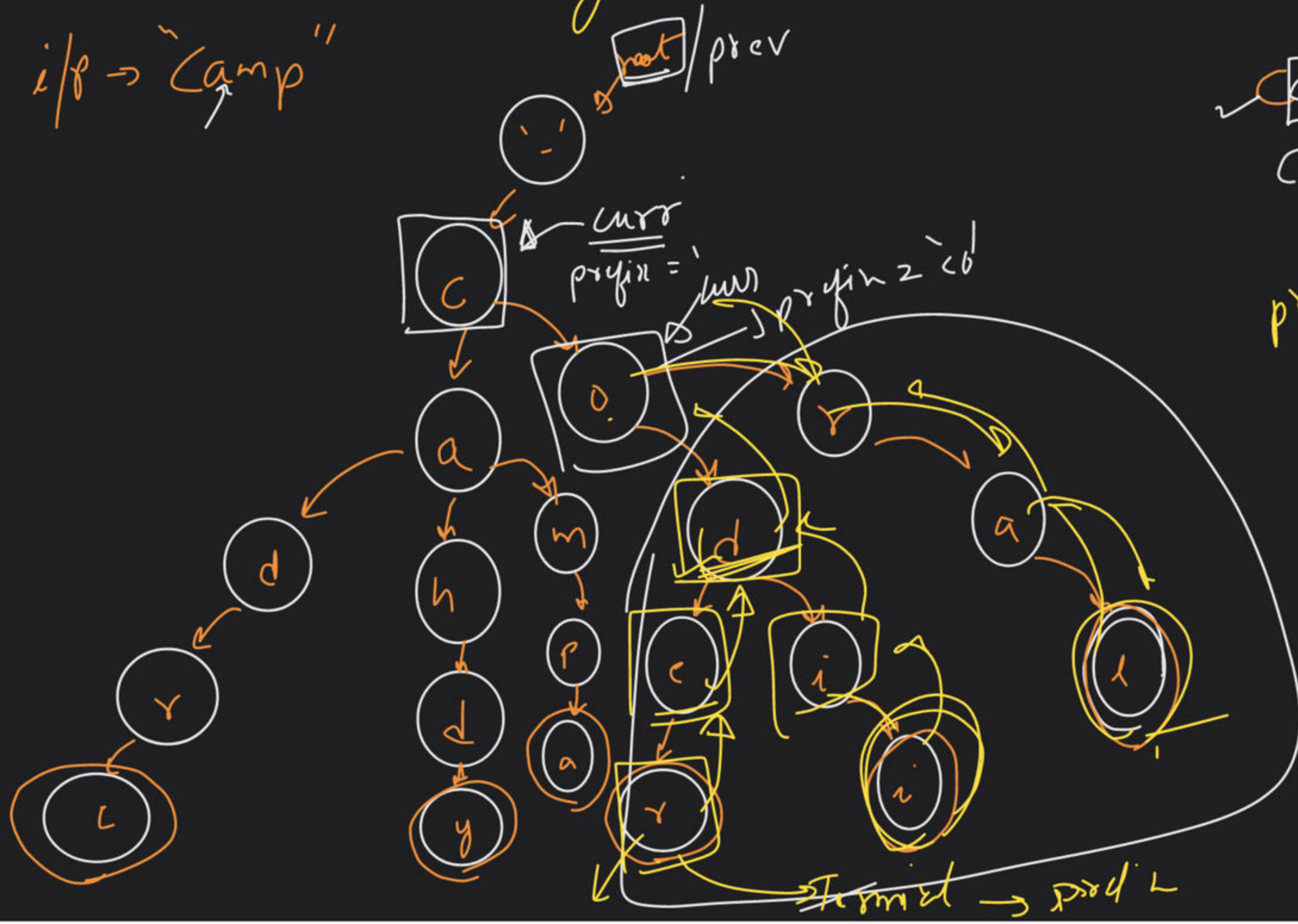
↓ $\boxed{0} \rightarrow 1011, 1002, 1001, 1000$
 ↑
 1010

$$\text{Cent ch}$$

lov \rightarrow lov₁, love, lov₂

lovi -> loviz

i/f \rightarrow "Camp"



Handwritten diagram illustrating the process of identifying a word. A box is drawn around the letter 'a', with a checkmark to its left and an arrow pointing to the right. Below this, another box is drawn around the letters 'am', with an arrow pointing to the right.

a →
am →

prefix = ~~code~~
ii
code
code

stop
stop

m-len str^y

→ trie

$$O(m \times n \times m)$$

$$O(nm^2)$$