Coin Change - Combinations - 1

Saturday, 31 July 2021 2:41 PM

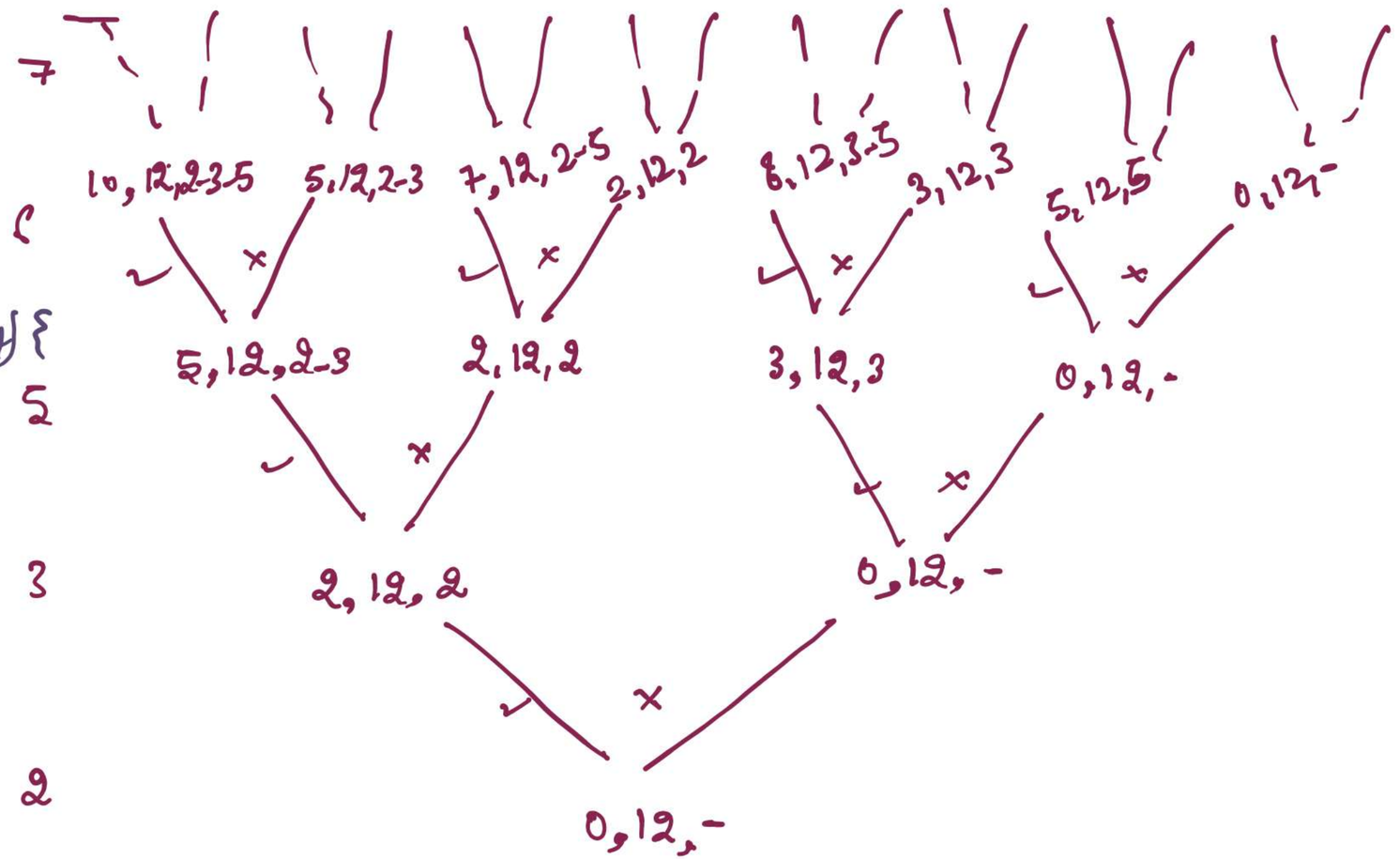
coins → 2, 3, 5, 6, 7 , target = 12 → 2_3_7
→ 5_7

finite coins

level → coins

option → choice of coins (yes or no)

```
if(indx >= coins.length){
    if( sumsofar == target){
        syso(asf);
    }
    return;
}
```



Coin Change - Combinations - 2

Saturday, 31 July 2021 2:41 PM

coins → 2, 3, 5, 6, 7   target = 12

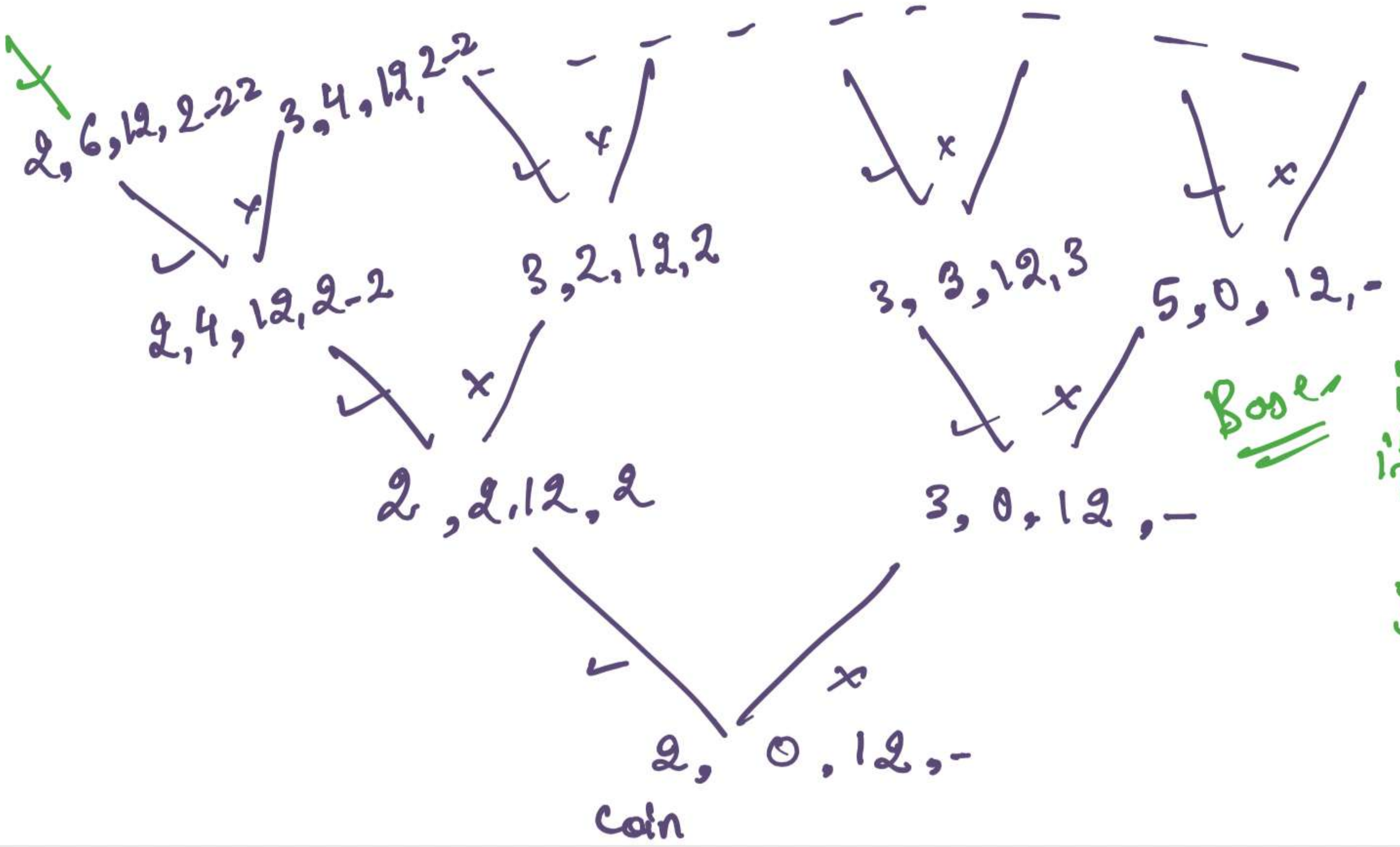
level → coins

option → Yes and No.

sumsofar + coins <= target → make Yes call

Yes → Next level is for same coin

No → Next level is for next coin



Requirement

1. Indx
2. coins
3. amtsf
4. target
5. asf

Base: if(amtsf == target) ans

if(indx == coins.len) return

}

Coin Change - Permutations - 1

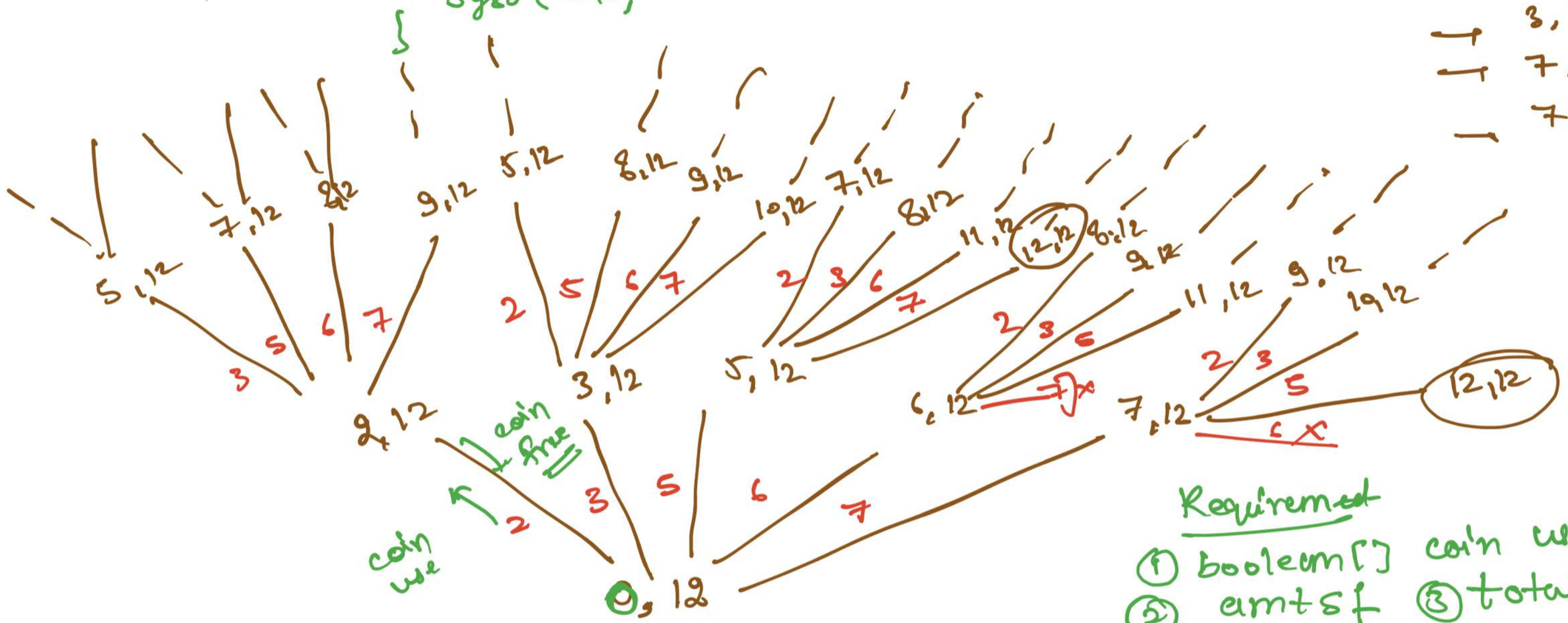
Saturday, 31 July 2021 2:41 PM

coins → 2, 3, 5, 6, 7 (finite coins), target = 12, permutations

- 2,3,7 → 5,7
- 2,7,3 → 7,5
- 3,2,7
- 3,7,2
- 7,2,3
- 7,3,2

Base cond'

```
if(amtsf == tamt){
    syso(asf);
}
```



Requirement

1. boolean[] coin used
2. amtsf
3. total amt
4. asf

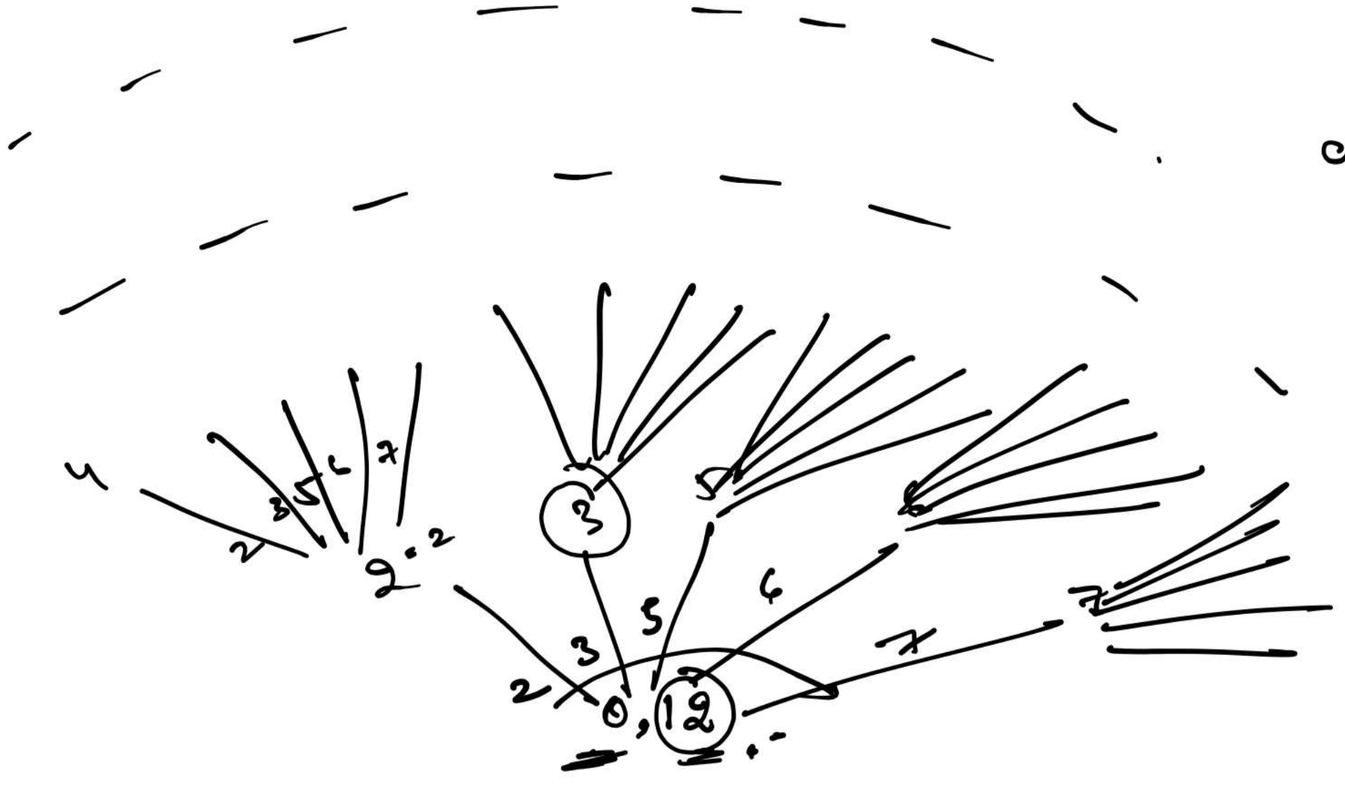
Coin Change - Permutations - 1

Saturday, 31 July 2021 2:41 PM

coins → {2, 3, 5, 6, 7} infinite , target = 12

Permute { 2-2-2-2-2-2 ; ; ;

Target=12



amtSf + Coin <= total amount

Requirment

1. amt so far
2. total amount
3. coins array
4. ans so far

Abbreviation Using Backtracking

Saturday, 31 July 2021 2:41 PM

String → str = "pep"

length of string = 3

No. of possible abbreviation = 2^{length}

$= 2^3 = 8$

0 → character

1 → as value

possible binary representation using 3 place				String's character
pep →	0	0	0	→ pep
pe1 →	0	0	1	→ pe1
p1p →	0	1	0	→ p1p
p2 →	0	1	1	→ p2
1ep →	1	0	0	→ 1ep
1e1 →	1	0	1	→ 1e1
2p →	1	1	0	→ 2p
3 →	1	1	1	→ 3

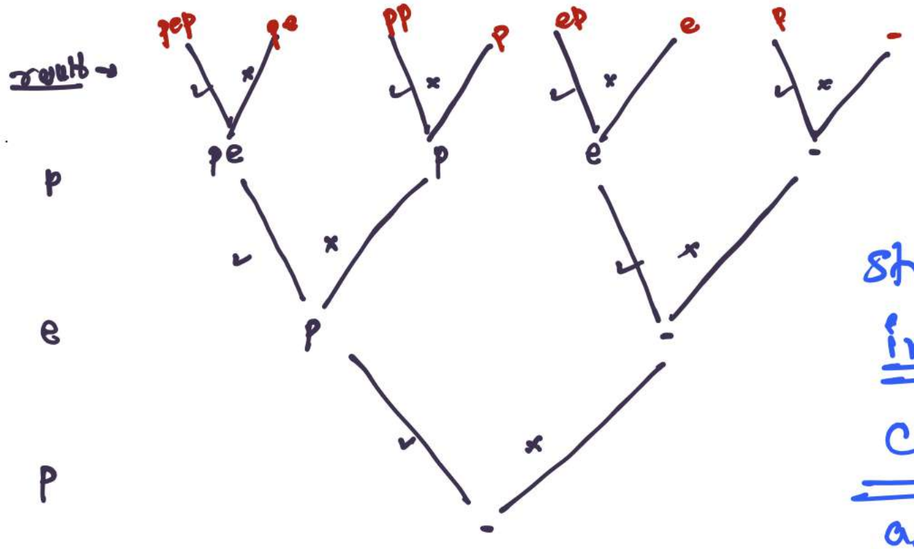
these are all abbreviation of pep

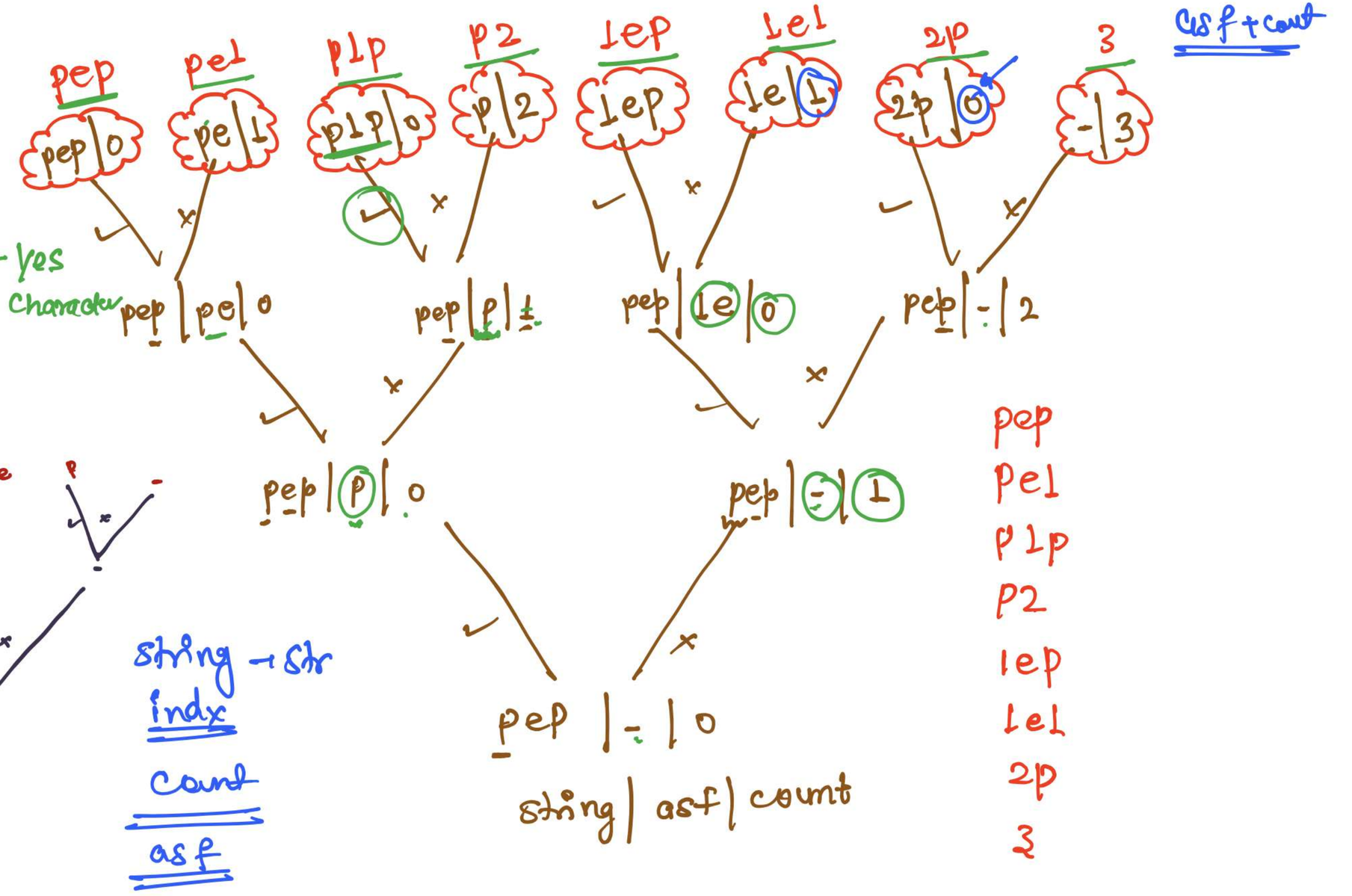
pep → level → character
option → choice of character

Yes call →
→ asf + count if ! = 0 + yes character

No call →
→ asf pass, count +1

count = 0





Max Score

Saturday, 31 July 2021 4:30 PM

4 ⟶ No. of words

dog cat dad good ⟶ 4 different words

9 ⟶ No. of characters

a b c d d d g o o ⟶ characters ⟶ Allowed Freq. of a character

⟶ freq. array

1 0 9 5 0 0 3 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0

a b c d e f g ... o ... z

Yes and No of word] No → ✗

Yes → freq remember] + score count

a → 1
b → 1
c → 1
d → 3
g → 1
o → 2

} Mapping of score associated with character

4 words ⟶ No. of possible subset of words with these words = 2^4

Ex. ① dog dad] ⟶ d → 3, o → 1, g → 1, a → 1 } these freq. are in valid Range] valid set

② dog dad good] ⟶ d → 3, o → 3, g → 2, a → 1 } our set is going beyond the range of available freq] Invalid set

⋮

Find max score from valid set??

Scrore ⟶ dog dad] ⟶ valid set ⟶ 5 + 2 + 3 + 5 + 1 + 5 = 21 } Maximise

dog cat dad good

abcdddgoo

level → words

option → choice of word

Yes and No

a → 1
b → 1
c → 1
d → 3
g → 1
o → 2

Requirement

1. freq array.
2. score array
3. score achieve
4. words array
5. index

Max { s1+s2, s1, s2+s3, s2, s3, 0 }

Max → cnt

global Max ⟵ to