**FP Growth Algorithm**

**TID |items\_bought**

T100 | {M, O, N, K, E, Y}

T200 | {D, O, N, K, E, Y}

T300 | {M, A, K, E}

T400 | {M, U, C, K, Y}

T500 | {C, O, O, K, I, E}

**Let the minimum support count =3**

**Solution**

**List out items and their support counts**

**Item | Support Count**

A | 1

C | 2

D |1

E |4

I |1

K |5

M |3

N |2

O |3

U |1

Y |3

**Generate L, but with the largest support count to smallest**

L = {{K: 5}, {E: 4}, {M: 3}, {O:3}, {Y: 3}}

**Item |Support Count**

K |5

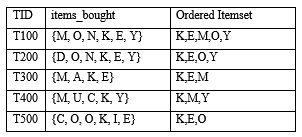
E |4

M |3

O |3

Y |3

So, rewriting the table with the new column:



Creating the tree

Steps

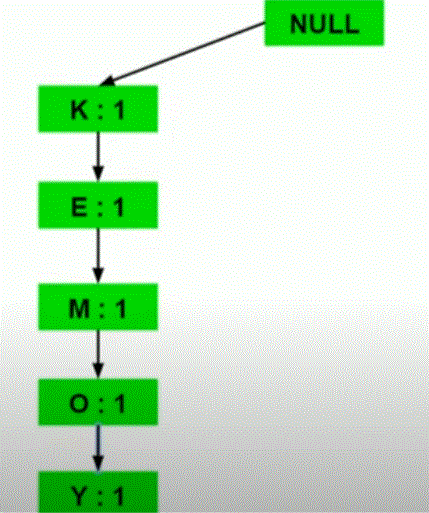
1. Write out table structure

2. Add empty node (null)

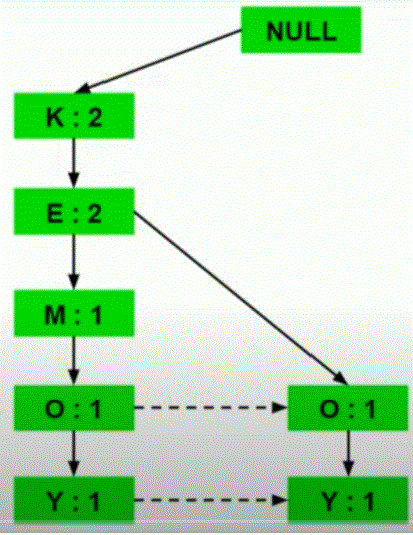
3. Go through each transaction’s ordered item-set and draw nodes and link them together. If the paths match, then add one to the count on the note itself.

All the ordered items are inserted into the Trie Data Structure

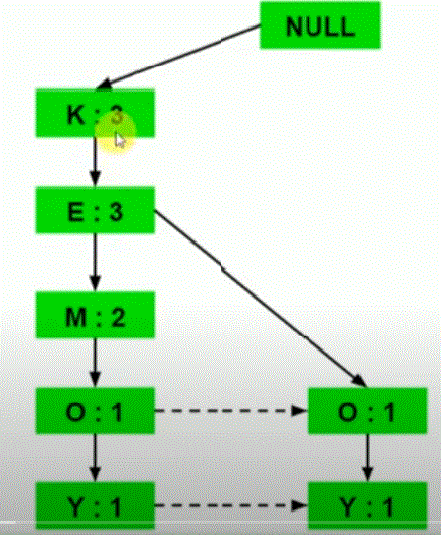
Inserting {KEMOY}



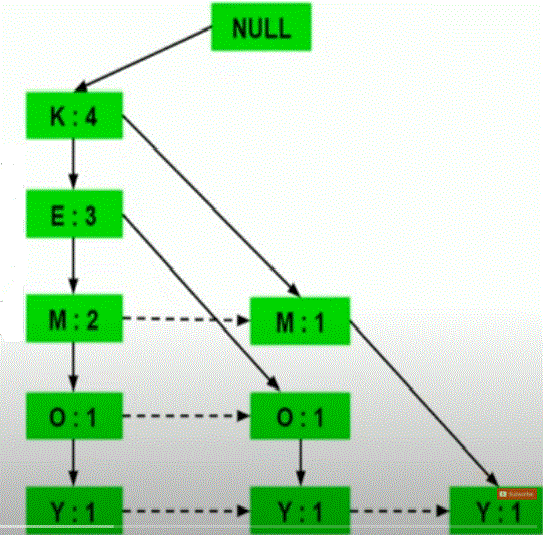
Inserting {KEOY}



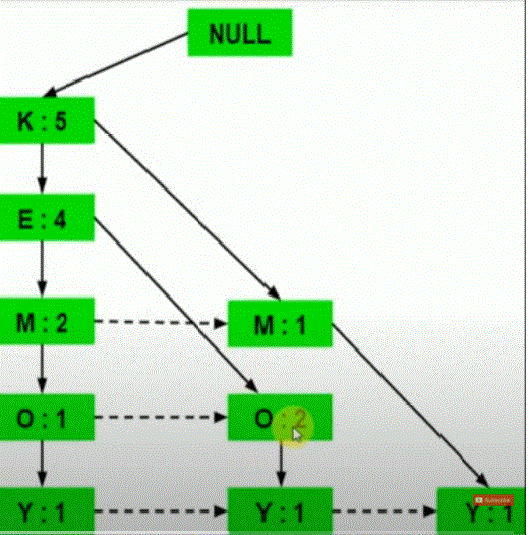
Inserting {KEM}

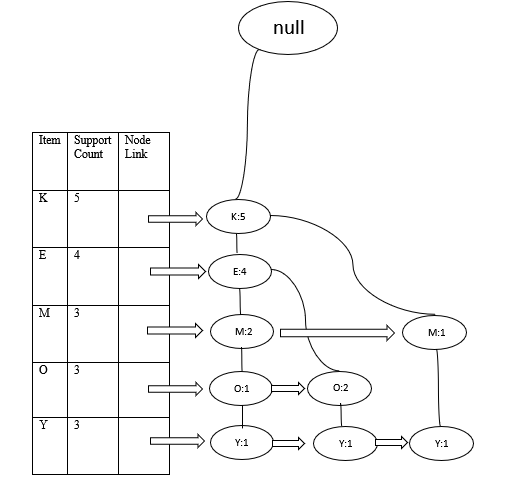


Inserting {KMY}



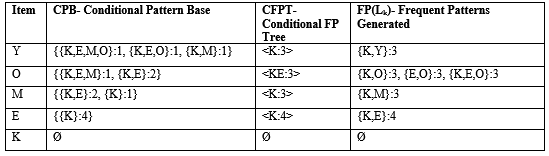
Inserting {KEO}





Find Conditional Pattern Base by analyzing all paths to an item.

Find Conditional FP Tree by finding common terms among record in Conditional Patter Base and summing its support count.



**Frequent Item-sets:**

L = { {K}:5, {E}:4, {M}:3, {O}:3, {Y}:3, {K,Y}:3, {K,O}:3, {E,O}:3, {K,E,O}:3, {K,M}:3, {K,E}:4 }