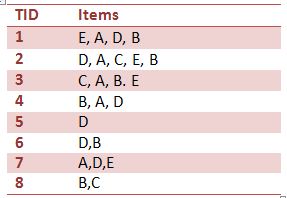
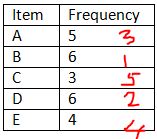
**Example Solved**

Q. Find all frequent itemsets or frequent patterns in the following database using FP-growth algorithm. Take minimum support as 30%.

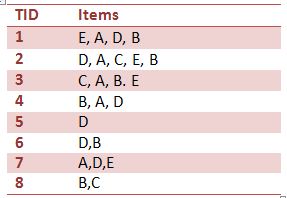


**Step 1 - Calculate Minimum support count**  
  
First we should calculate the minimum support count. Question says minimum support should be 30%. It calculate as follows:  
  
Minimum support count(30/100 \* 8) = **2.4**  
  
As a result, 2.4 appears but to empower the easy calculation it can be rounded to to the ceiling value. Now,   
  
Minimum support count is **ceiling**(2.4) = **3**

 **Step 2 - Find frequency of occurrence**  
Now time to find the frequency of occurrence of each item in the Database table. For example, item A occurs in row 1,row 2,row 3,row 4 and row 7. Totally 5 times occurs in the Database table. You can see the counted frequency of occurrence of each item in Table 2.

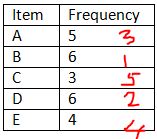
[](http://2.bp.blogspot.com/-YvSVN54CJaI/Tghx7excdzI/AAAAAAAAAG0/M1QRK4JIsDk/s1600/2.JPG)

|  |
| --- |
|  |
| Table2 -Frequency of Occurrence |

  
 **Step 3 - Prioritize the items**  
In Table 2 you can see the numbers written in Red pen. Those are the priority of each item according to it's frequency of occurrence. Item B got the highest priority (**1**) due to it's highest number of occurrences. At the same time you have opportunity to drop the items which not fulfill the minimum support requirement.For instance, if Database contain **F**which has frequency 1, then you can drop it.

\*Some people display the frequent items using list instead of table. The frequent item list for the above table will be **B:6, D:6, A: 5, E:4, C: 3**.  
  
**Step 4 -Order the items according to priority**  
As you see in the Table 3 new column added to the Table 1. In the Ordered Items column  all the items are queued according to it's priority, which mentioned in the Red ink in Table 2. For example, in the case of ordering  row 1, the highest priority item is B and after that D, A and E respectively.

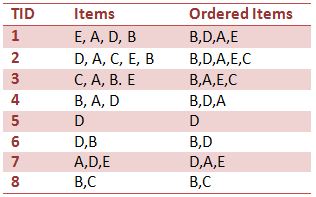
|  |
| --- |
| <http://3.bp.blogspot.com/-foobGsFih_w/TghzmkvHdHI/AAAAAAAAAG4/R-UVc2-6NHM/s1600/3.JPG> |
| Table 3 - New version of the Table 1 |

[](http://2.bp.blogspot.com/-YvSVN54CJaI/Tghx7excdzI/AAAAAAAAAG0/M1QRK4JIsDk/s1600/2.JPG)  
  
**Step 5 –Now we can start designing fp =tree**  
As a result of previous steps we got a ordered items table (Table 3). Now it's time to draw the FP tree. I'll mention it row by row.

**Row 1:**

Note that all FP trees have 'null' node as the root node. So draw the root node first and attach the items of the  row 1 one by one respectively. (See the Figure 1) And write their occurrences in front of it. (write using a pencil ,because next time we have to erase it.)

|  |
| --- |
| [http://2.bp.blogspot.com/-rQkSVlCfVpI/Tgh5DS3KR1I/AAAAAAAAAG8/6mIiH8G7EuY/s320/4.JPG](http://2.bp.blogspot.com/-rQkSVlCfVpI/Tgh5DS3KR1I/AAAAAAAAAG8/6mIiH8G7EuY/s1600/4.JPG) |
| Figure  1- FP tree for  Row 1 |

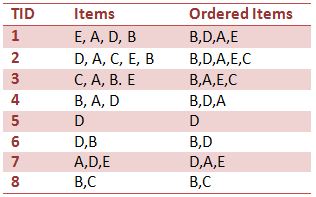
[](http://3.bp.blogspot.com/-foobGsFih_w/TghzmkvHdHI/AAAAAAAAAG4/R-UVc2-6NHM/s1600/3.JPG) **Row 2:**  
Then update the above tree (Figure 1) by entering the items of row 2. The items of row 2 are B,D,A,E,C. Then without creating another branch you can go through the previous branch up to E and then you have to create new node after that for C. This case same as a scenario of traveling through a road to visit the towns of the country. You should go through the same road to achieve another town near to the particular town.  
When you going through the branch  second time you should erase one and write two for indicating the two times you visit to that node.If you visit through three times then write three after erase two. Figure 2 shows the FP tree after adding row 1 and row 2. Note that the red underlines which indicate the traverse times through the each node.

|  |
| --- |
| [http://3.bp.blogspot.com/-Uv1-jJxpwHI/Tgh7psRcRXI/AAAAAAAAAHA/FNQkgWj9UXc/s320/5.JPG](http://3.bp.blogspot.com/-Uv1-jJxpwHI/Tgh7psRcRXI/AAAAAAAAAHA/FNQkgWj9UXc/s1600/5.JPG) |
| Figure  2- FP tree for Row 1,2 |

**Row 3:**  
In row 3 you have to visit B,A,E and C respectively.  So you may think you can follow the same branch again by replacing the values of B,A,E and C . But you can't do that you have opportunity to come through the  B. But  can't connect B to existing A overtaking D. As a result you should draw another A and connect it to B and then connect new E to that A and new C to new E. See Figure 3.

**Row 4:**  
Then row 4 contain B,D,A. Now we can just rename the frequency of occurrences in the existing branch. As B:4,D,A:3.  
 **Row 5:**

|  |
| --- |
| [http://2.bp.blogspot.com/-xemdlKBz_24/TgozShdLnvI/AAAAAAAAAHM/m4bRtsn-vsQ/s320/1.JPG](http://2.bp.blogspot.com/-xemdlKBz_24/TgozShdLnvI/AAAAAAAAAHM/m4bRtsn-vsQ/s1600/1.JPG) |
| Figure 3 - After adding third row |

[](http://3.bp.blogspot.com/-foobGsFih_w/TghzmkvHdHI/AAAAAAAAAG4/R-UVc2-6NHM/s1600/3.JPG)

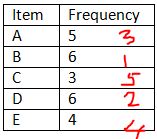
In fifth raw have only item D. Now we have opportunity draw new branch from 'null' node. See Figure 4.

|  |
| --- |
| [http://1.bp.blogspot.com/-kNRx4oy3450/Tgo1lTYEfNI/AAAAAAAAAHQ/IrsNlC0YuXs/s320/2.JPG](http://1.bp.blogspot.com/-kNRx4oy3450/Tgo1lTYEfNI/AAAAAAAAAHQ/IrsNlC0YuXs/s1600/2.JPG) |
| Figure 4- Connect D to null node |

**Row 6:**  
B and D appears in row 6. So just change the B:4 to B:5 and D:3 to D:4.

**Row 7:**  
Attach two new nodes A and E to the D node which hanging on the null node. Then mark D,A,E as D:2,A:1 and E:1.  
 **Row 8 :(Ohh.. last row)**  
Attach new node C to B. Change the traverse times.(B:6,C:1)

|  |
| --- |
| [http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s320/6.JPG](http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s1600/6.JPG) |
| Figure 5 - Final FP tree |

[](http://2.bp.blogspot.com/-YvSVN54CJaI/Tghx7excdzI/AAAAAAAAAG0/M1QRK4JIsDk/s1600/2.JPG) **Step 6 – Validation**  
After the five steps the final FP tree as follows: Figure 5.  
How we know is this correct?  
Now count the frequency of occurrence of each item of the FP tree and compare it with Table 2. If both counts equal, then it is positive point to indicate your tree is correct.

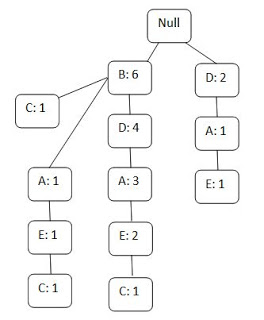
**Part 2**

Now, for each item, the Conditional Pattern Base is computed which is path labels of all the paths which lead to any node of the given item in the frequent-pattern tree. Note that the items in the below table are arranged in the ascending order of their frequencies.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| <http://2.bp.blogspot.com/-YvSVN54CJaI/Tghx7excdzI/AAAAAAAAAG0/M1QRK4JIsDk/s1600/2.JPG> | |  |  | | --- | --- | | Items | Conditional pattern Base | | C | {B:1} {B,A,E:1} {B,D,A,E:1} | | E | {B,A:1} {B,D,A:2} {D,A:1} | | A | {B:1} {B,D:3} {D:1} | | D | {B:4} | | B | Cant consider as B is root element | | [http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s320/6.JPG](http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s1600/6.JPG) |

Now for each item the Conditional Frequent Pattern Tree is built. It is done by taking the set of elements which is common in all the paths in the Conditional Pattern Base of that item and calculating it’s support count by summing the support counts of all the paths in the Conditional Pattern Base.

Minimum support count is  **3**

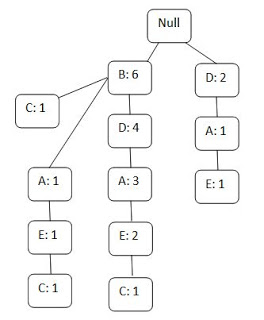
[](http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s1600/6.JPG)

|  |  |
| --- | --- |
| Items | Conditional pattern Base |
| C | {B:1} {B,A,E:1} {B,D,A,E:1} |
| E | {B,A:1} {B,D,A:2} {D,A:1} |
| A | {B:1} {B,D:3} {D:2} |
| D | {B:4} |
| B | Cant consider for root element |

|  |  |
| --- | --- |
| Items | Conditional pattern Base |
| C | { B:3} |
| E | {B:3,A:3} |
| A | {B:4,D:3} |
| D | {B:4} |
| B | Cant consider for root element |

From the Conditional Frequent Pattern tree, the **Frequent Pattern rules** are generated by pairing the items of the Conditional Frequent Pattern Tree set to the corresponding to the item as given in the below table.

|  |  |
| --- | --- |
| Items | Conditional pattern Base |
| C | {B,C:3} |
| E | {B,E:3},{A,E:3},{B,A,E:3} |
| A | {B,A:4},{D,A:3} ,{B,D,A:3} |
| D | {B,:4} |
| B | Cant consider for root element |

[](http://2.bp.blogspot.com/-DkbVUxHRxu0/Tgh-J1H3PnI/AAAAAAAAAHE/h0kAGVtaXfI/s1600/6.JPG)

|  |  |
| --- | --- |
| Items | Conditional pattern Base |
| C | {B,C:3} |
| E | {B,E:3},{A,E:3},{B,A,E:3} |
| A | {B,A:4},{D,A:3} ,{B,D,A:3} |
| D | {B,:4} |
| B | Cant consider for root element |

Rules From the Above Tables are

Consider Confidence as 50%

**Confidence(A->B)=Support\_count(A∪B)/Support\_count(A)**

**{B}  C conf(BC)=S.C(BUC)/SC(B) 3/3=100%**

**{B}E 3/3=100%**

**{A}E**

**{B,A}E**

**{B}A**

**{D}A**

**{B,D}A ¾\*100=75%**

**{B}D**