A Numeric Approach to Mine Frequent Patterns From Very Large Database

A dissertation submitted in the partial fulfillment for the award of Degree of

Master of Technology

In

Software Engineering

Submitted by

Sachin Mittal (2K12/SWT/17)

COMPUTER ENGINEERING DEPARTMENT

DELHI TECHNOLOGICAL UNIVERSITY

BAWANA ROAD, DELHI

July 2015

DECLARATION

I hereby declare that the thesis entitled "**A numeric approach to mine frequent patterns from very large database**" which is being submitted to the Delhi Technological University, in partial fulfillment of the requirements for the award of degree of Master of Technology in Software Technology is an authentic work carried out by me. The material contained in this thesis has not been submitted to any university or institution for the award of any degree.

Sachin Mittal
2K12/SWT/17
Master of Technology (Software Technology)
Delhi Technological University
Bawana road, Delhi - 110042

CERTIFICATE

Date: __________

This is to certify that the Major Project entitled "**A numeric approach to mine frequent patterns from very large database"** submitted by **SACHIN MITTAL**, Roll Number: **2K12/SWT/17**; in partial fulfillment of the requirement for the award of degree Master of Technology in Software Technology to Delhi Technological University, Bawana Road Delhi; is a record of the candidate's own work carried out by him under my supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other Degree.

Dr. Kapil Sharma
Asst. Professor, Computer Engineering Dept.
Delhi Technological University
Bawana road, Delhi – 110042

ACKNOWLEDGEMENT

July 2015

I would like to take this opportunity to thank my project guide Dr. KAPIL SHARMA for his invaluable time and consistent guidance during this whole research work. I am very thankful to him for giving me the opportunity and motivation to complete this challenge. I am very appreciative of his generosity with his time, advice, data, and references, to name a few of his contributions. Under his able guidance I was enabled to achieve the objectives of this work.

Lastly, I would like to thank all the people who were involved directly or indirectly in successfully completion of this project.

Sachin Mittal
2K12/SWT/17
Master of Technology (Software Technology)
Delhi Technological University
Bawana road, Delhi – 110042

CONTENT TABLE

TABLE LIST

FIGURE LIST

ABSTRACT

Mining Frequent pattern is one of the major activities in Data Mining field to extract the useful information from large databases. Today is era of big shopping complexes, mega stores, super-markets. Shopping using e-commerce portals like flipkart, amazon, snapdeal etc is increasing day by day. As a result databases of all these mega stores, shopping complexes and e-commerce portals are increasing many folds every year. Frequent patterns are used by the big corporate houses to know the interest and purchasing trend of their customers and accordingly they plan their sales or marketing strategies. This lead to need for faster algorithm to mine frequent patterns from these large databases.

This research work presents a novel idea for mining frequent patterns from very large database. Proposed new algorithms i.e. Magic Number Algorithm is based on the converting the database items to Numeric equivalents. Once we have all items represented by numerals then we can apply mathematics and logic on them in easier and faster way as compared to operations on strings.

Using same experimental setup and environment conditions, proposed new algorithm proves superior to Apriori Algorithm by approximate 70% performance improvement.

At Minimum Support level = 4 and transaction count= 1000,000
Time taken by Magic Number Algorithm - **105 milli seconds**
Time taken by Apriori Algoritm - **345 milli seconds**

Development Environment Detail:

OS:	Android 5.0 (Lollypop)
SDK:	ADT Build: v21.1.0-569685
CPU:	ARM, 1.9GHz Quad Core
RAM:	3 GB
API:	Level 17

Chapter 1

1. INTRODUCTION

Proposed new algorithm deals with Data Mining from large databases. Active work on mining useful knowledge and information from very large database started in 90's [1] [2]. Frequent patterns are set of data items which occur more than a given threshold value in given large database set. Mining of frequent patterns is considered one of the most critical & important concepts in data mining because this tells about the trend of occurrence of data items. For example in large grocery stores this can tell about liking of customer for one data item as compared to other. By mining frequent patterns business houses can predict what kind of items needs to be presented to customer to increase the sale volume. Similarly in other areas also like scientific research, Universities data frequent patterns can be used to mine various useful facts for future strategy design.

Very large databases have millions of records and it is not possible to read all that information and extract the useful information. When strategists from various domains sit to analyze the available information, he need to know the trend of customers and users to design his strategy to increase the volume of sales etc. Frequent Patterns are most useful tool to help strategists to design their strategy. Frequent patters gives direct insight about users preferences and purchase habits in a particular domain of business like grocery store, electronics mega stores, e-commerce etc.

1.1 Existing Popular Algorithm

Some of existing algorithms for frequent pattern mining are:

Apriori:

Apriori Algorithm [3] is based on making larger and larger Item sets. It works by identifying the frequently occurring individual items in the database and making larger sets from them which are as long as those item sets that appear in the database more than required threshold. Such frequent item sets detected by Apriori are further used to define rules of association to know the ongoing trend.

FP Growth:

FP growth approach [4] is based on data structure FP-Tree for storing crucial but compressed information regarding frequent patterns. This approach smartly avoids costly candidate generation of Ariori. This was further elaborated in journal "Data Mining and Knowledge Discovery" [5]. It executes in multi passes. While in very first pass this counts occurrence frequency of items (attribute-value pairs) in the dataset, and use ‘header table’ to store them. In next pass, FP-tree structure is built by inserting instances. Items in each instance are sorted by descending order of their occurrence count in the dataset to process tree quickly. Items are discarded in each instance on not meeting minimum coverage threshold. FP-tree provides high compression near to root of tree If many instances share most frequent item. Final dataset is mined after recursive processing of this compressed growth tree.

1.2 Goal of the thesis

The goal of the work in this thesis is summarized below:

- Go through about existing/past work in Frequent patterns mining area
- Explanation of new algorithm using Pseudo code, Flow Chart, real life example
- Checking Algorithm accuracy and performance using experimental setup
- Comparison with well-known and widely used Apriori algorithm

1.3 Organization of the Thesis

The thesis is organized as follows:

Chapter 2 gives overview regarding work done in past related to frequent patterns.

Chapter 3 explain most basic and popular algorithm in detail

Chapter 4 discusses about new proposed algorithm. Its pseudo code, flow chart and explanation using real life example.

Chapter 5 checks about accuracy of proposed algorithm using experimental setup. Description of experimental setup’s hardware & software detail. Comparison of execution performance with well-known widely used algorithm i.e. Apriori Algorithm

Chapter 6 presents the conclusions of the thesis and future work.

Chapter 2

2. RELATED WORK

R. Agrawal and R. Srikant et al. [3] in 1993, in their paper “Fast Algorithms for Mining Association Rules”, described the Apriori and AprioriTid Algorithm. Apriori is based on keep making the larger Item sets. It identifies the frequent individual items in the database. Further it extends them to larger item sets as till those item-sets appear in the database more than set threshold. The frequent item sets detected using Apriori are utilized to determine association rules. These rules give indication about ongoing general trends in the database.

H. Maurice and A. Swami, et al. [6] described Method of mining frequent patterns using Sets where mining was performed on sets of various transactions.

J. Han, J. Pei and Y. Yin, et al. [4] described Mining Frequent Patterns without Candidate Generation. This had advantage over Apriori Algorithm which require repeated candidate set generation for frequent patterns generation.

J. Han, J. Pei and Y. Yin, et al. [5] in 2004 described frequent-pattern tree (FP-tree) structure, as an extended prefix-tree structure for storing compressed but crucial information related to frequent patterns and further designing an efficient FP-tree based method of mining ie FP-growth, for mining the complete set of frequent patterns by pattern fragment growth.

A. Raorane, R. Kulkarni and B. Jitkar et al. [7] explained to mine frquqent patterns using Market Basket Analysis

T. Patel, M. Panchal, D. Ladumor, J. Kapadia, P. Desai, A. Prajapati and R. Prajapati et al. [8] did survey of various methods to mine frequent patterns and provided analystical study result.

L. Wang, D. Cheung, R. Cheng, S. D. Lee and X. S. Yang, et al. [9] explained Mining of frequent item sets on uncertain databases. In uncertain databases, the support of an itemset is a random variable instead of a fixed occurrence counting of this itemset. Recently, with many new applications, such as sensor network monitoring, moving object search and network analysis uncertain data mining has become a hot topic in data mining.

R. K. Ahir and M. B. Ahir, et al. [10] provided comparattive study of various Frequent Patterns techniques with advantage and disadvantages with each other.

S. Zhang, Z. Du and J. Wang, et al. [11] explained Techniques for Mining Frequent Patterns in Unordered Trees aiming to discover restrictedly embedded sub tree patterns from a set of rooted labeled unordered trees.

Chapter 3

3. POPULAR ALGORITHMS DETAIL

Several researches have been done in past especially since 1990's for mining frequent patterns and association rules from VLDB.

R. Agrawal and R. Srikant et al. [3], in their paper "Fast Algorithms for Mining Association Rules", described the Apriori and AprioriTid Algorithm. Apriori is based on keep making the larger Item sets. It identifies the frequent individual items in the database. Further it extend them to larger item sets as till those item-sets appear in the database more than set threshold. The frequent item sets detected using Apriori are utilized to determine association rules. These rules give indication about ongoing general trends in the database.

The AprioriTid algorithm differs from classical Apriori that the database is not used at all for counting the support of candidate itemsets after the first pass. Rather, an encoding of the candidate itemsets used in the previous pass is employed for this purpose. In later passes, the size of this encoding can become much smaller than the database, thus saving much reading effort as compared to Apriori Algorithm

3.1 Apriori Algorithm

Apriori pruning principle:

If any itemset which is not frequent, its superset should not be generated/tested! R. Agrawal and R. Srikant et al. [3] described algorithm as below:

Method:

- Initially, one time scan of DB to fetch frequent 1-itemset
- Generate length (k+1) candidate itemsets from length k frequent itemsets
- Test the candidates against DB
- Terminate when no frequent or candidate set can be generated

Pseudo-code:

```
$C_k$: Candidate itemset of size k
$L_k$ : frequent itemset of size k
$L_1$ = {frequent items};
for ($k = 1$; $L_k \neq \emptyset$; $k$++) do begin
$C_{k+1}$ = candidates generated from $L_k$;
for each transaction $t$ in database do
    increment the count of all candidates in $C_{k+1}$
    that are contained in $t$
$L_{k+1}$ = candidates in $C_{k+1}$ with min_support
end
return $\cup_k L_k$;
```

Example of Candidate-generation

Suppose L_3=*{abc, abd, acd, ace, bcd}*

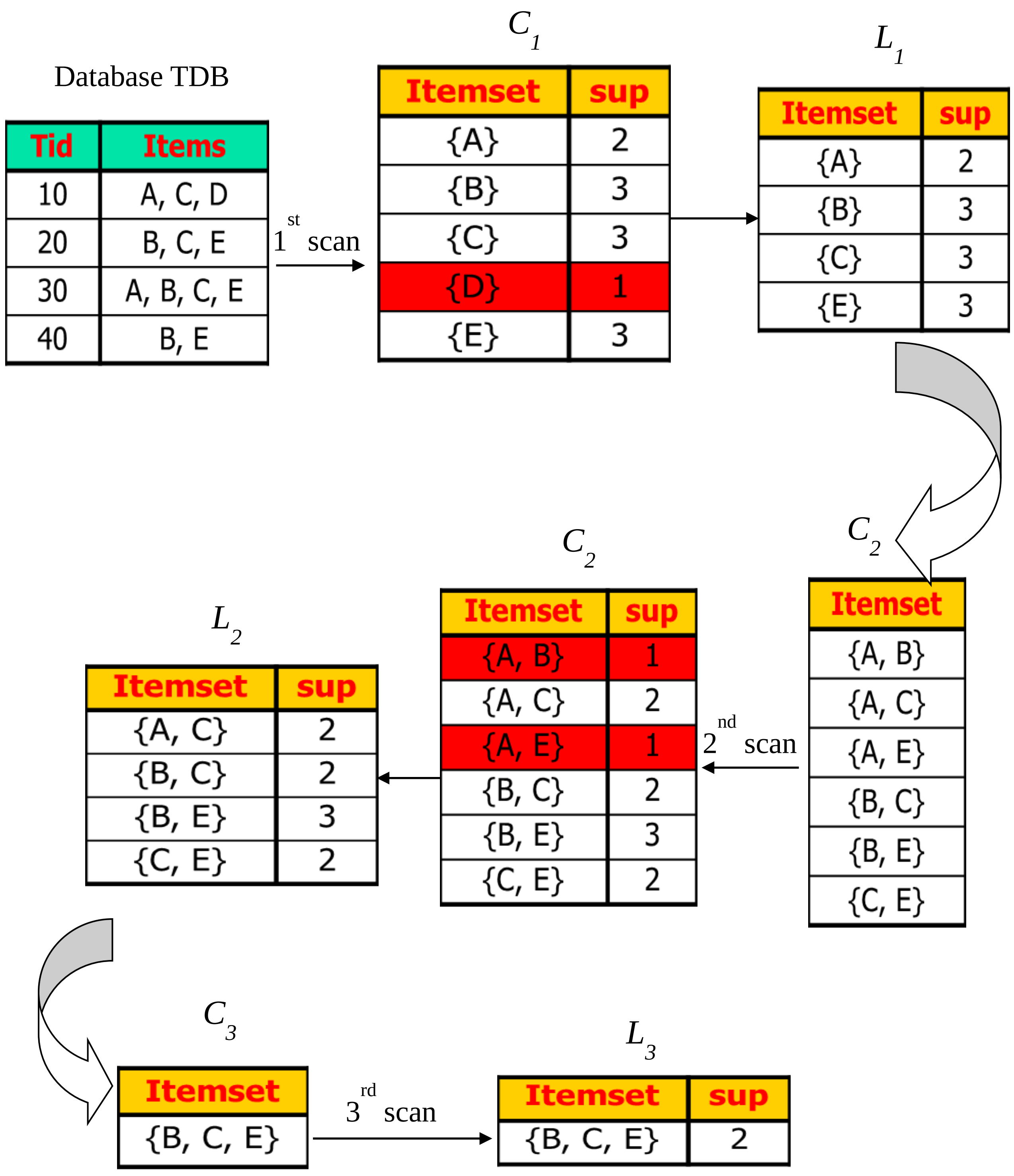
- Self-joining: $L_3 * L_3$
  - *abcd* from *abc* and *abd*
  - *acde* from *acd* and *ace*
- Pruning:
  - *acde* is removed because *ade* is not in L_3
- C_4={*abcd*} /* C_4 Candidate generated */

Example:

min_support = 2

Tid	**Items**
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

Table 3.1 Apriori Transactional Database



Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

1st scan

C_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

C_2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

2nd scan

C_2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

L_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

C_3

Itemset
{B, C, E}

3rd scan

L_3

Itemset	sup
{B, C, E}	2

$Sup_{min} = 2$

3.2 AprioriTid

R. Agrawal and R. Srikant et al. [3], in their paper "Fast Algorithms for Mining Association Rules", described the AprioriTid Algorithm. The best feature of this algorithm is that the database D is not used for counting support after the first pass. Rather, the set C_k is used for this purpose. Each member of the set C_k is of the form < TID, { X_k } >, where each X_k is a Potentially large k-item & present in the transaction with identifier TID. For $k = 1$, C_1 corresponds to the database D.

<u>Deficiencies of Apriori Algorithm:</u>

- Very big number of candidates
- Lot many scans of transaction database
- Too much workload of support counting for generated candidates

3.3 FP Growth Tree

FP tree is Mining Frequent Patterns technique without Candidate Generation.

J. Han, J. Pei and Y. Yin, et al. [4] , described frequent-pattern tree (FP-tree) structure, as an extended prefix-tree structure for storing compressed but crucial information related to frequent patterns and further designing an efficient FP-tree based method of mining ie FP-growth, for mining the complete set of frequent patterns by pattern fragment growth.

Efficiency of mining is achieved with three techniques:

1. Compression of large database into a smaller data structure ie. FP-tree which avoids costly and repeated scans of database.
2. It prefer a pattern-fragment method of growth to avoid the costly generation of a large number of candidate sets
3. a divide-and-conquer method based on partitioning is used to decompose the mining task in set of various smaller tasks for mining confined patterns in conditional databases, This result in reducing reduces the search space to large extent.

Various researches work shows that this method is efficient and scalable for mining both long and short frequent patterns, and is faster than the Apriori algorithm. It is claimed that it is faster than other frequent-pattern mining methods.

3.4 FP Growth Tree Example:

min-sup=2

TID	**Items**	**TID**	**Items**	**TID**	**Items**
T100	I1,I2,I5	T400	I1,I2,I4	T700	I1,I3
T200	I2,I4	T500	I1,I3	T800	I1,I2,I3,I5
T300	I2,I3	T600	I2,I3	T900	I1,I2,I3

Table 3.2 FP Tree Transactional Database

- The first scan of data is the same as Apriori
- Derive the set of frequent 1-itemsets
- Generate a set of ordered items

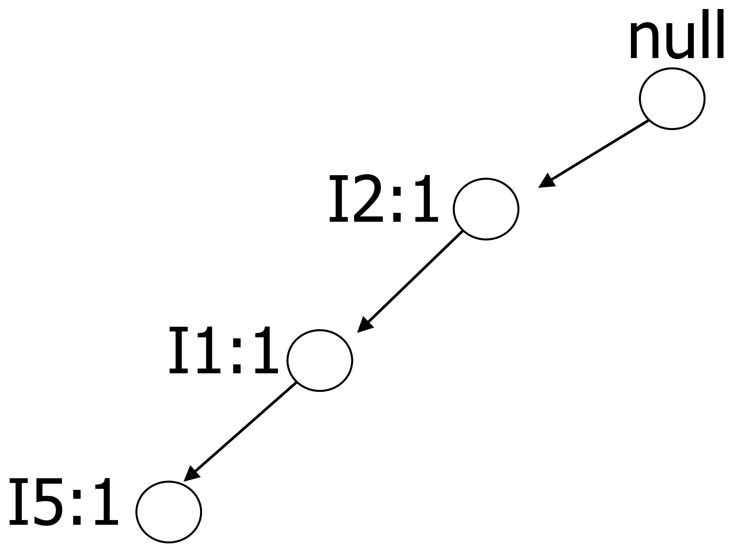
Item ID	Support count
I2	7
I1	6
I3	6
I4	2
I5	2

Table 3.3 FP Tree Database With Support Count

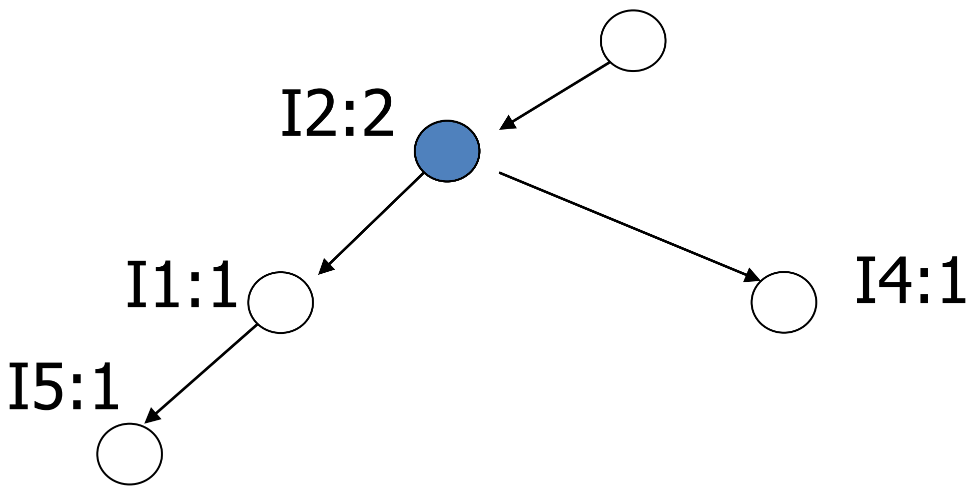
Construct the FP-Tree

- Create a branch for each transaction
- Items in each transaction are processed in order
- Item having maximum support need to be nearest to root

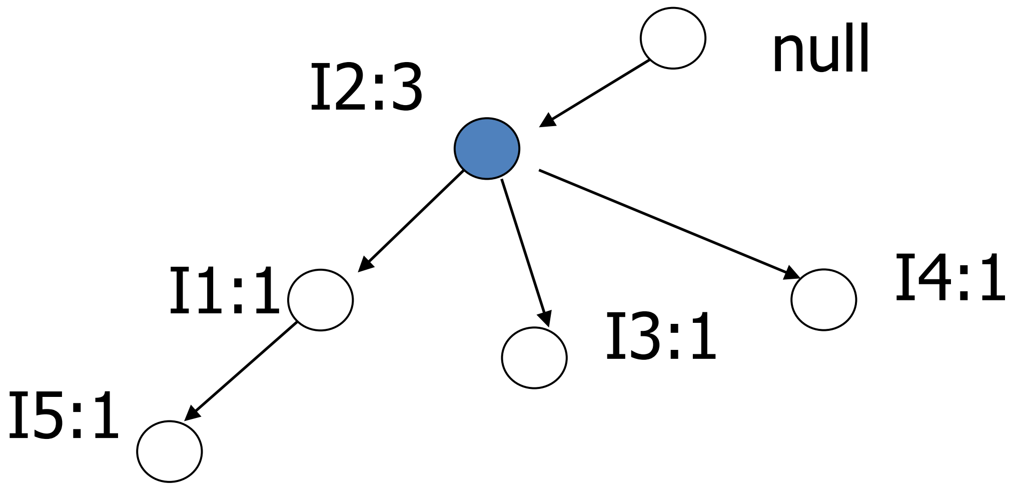
Order the items and create below branch for transaction T100: {I2,I1,I5}



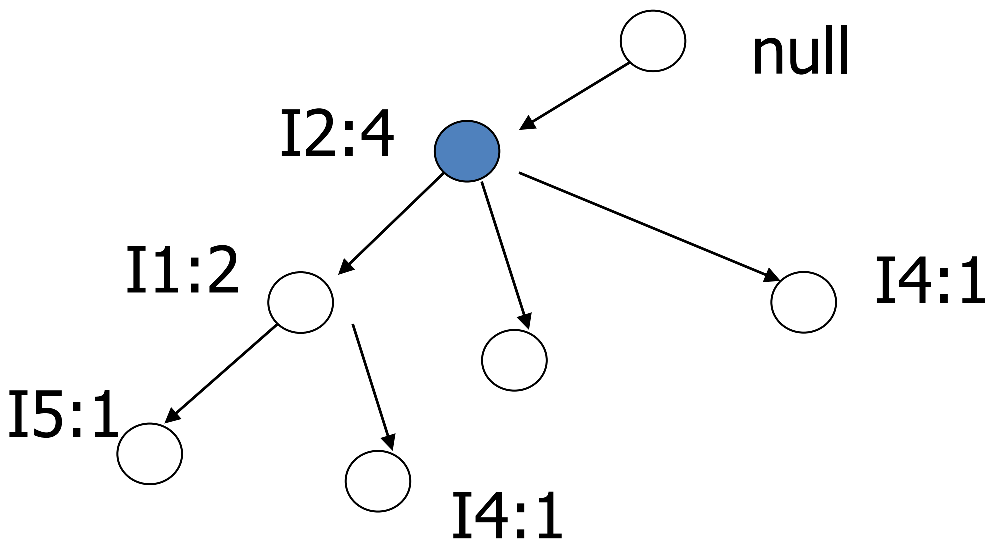
Order the items and create below branch for transaction T200: {I2,I4}



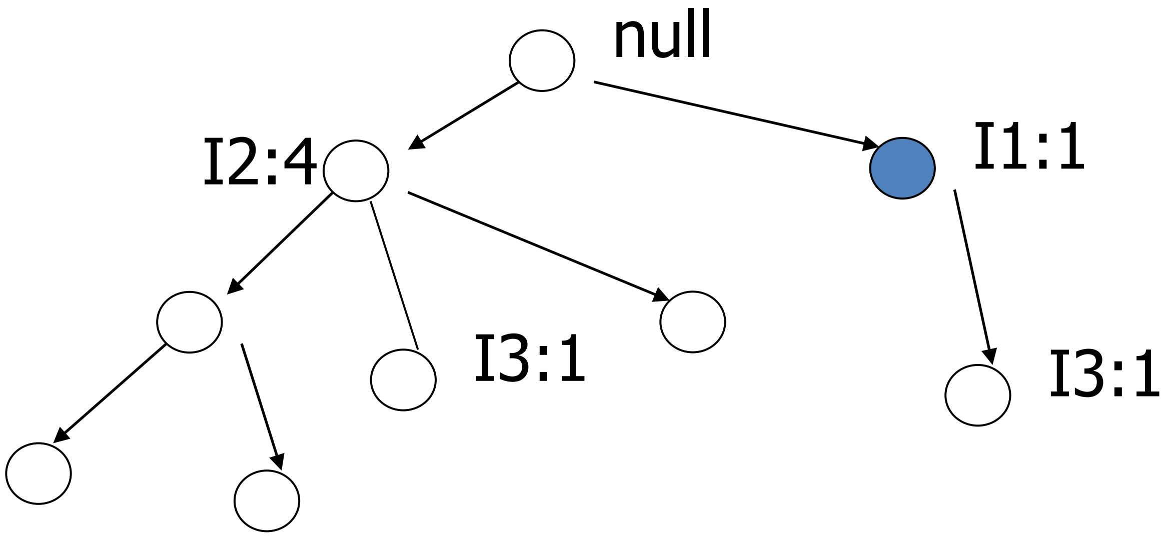
Order the items and create below branch for transaction T300: {I2,I3}



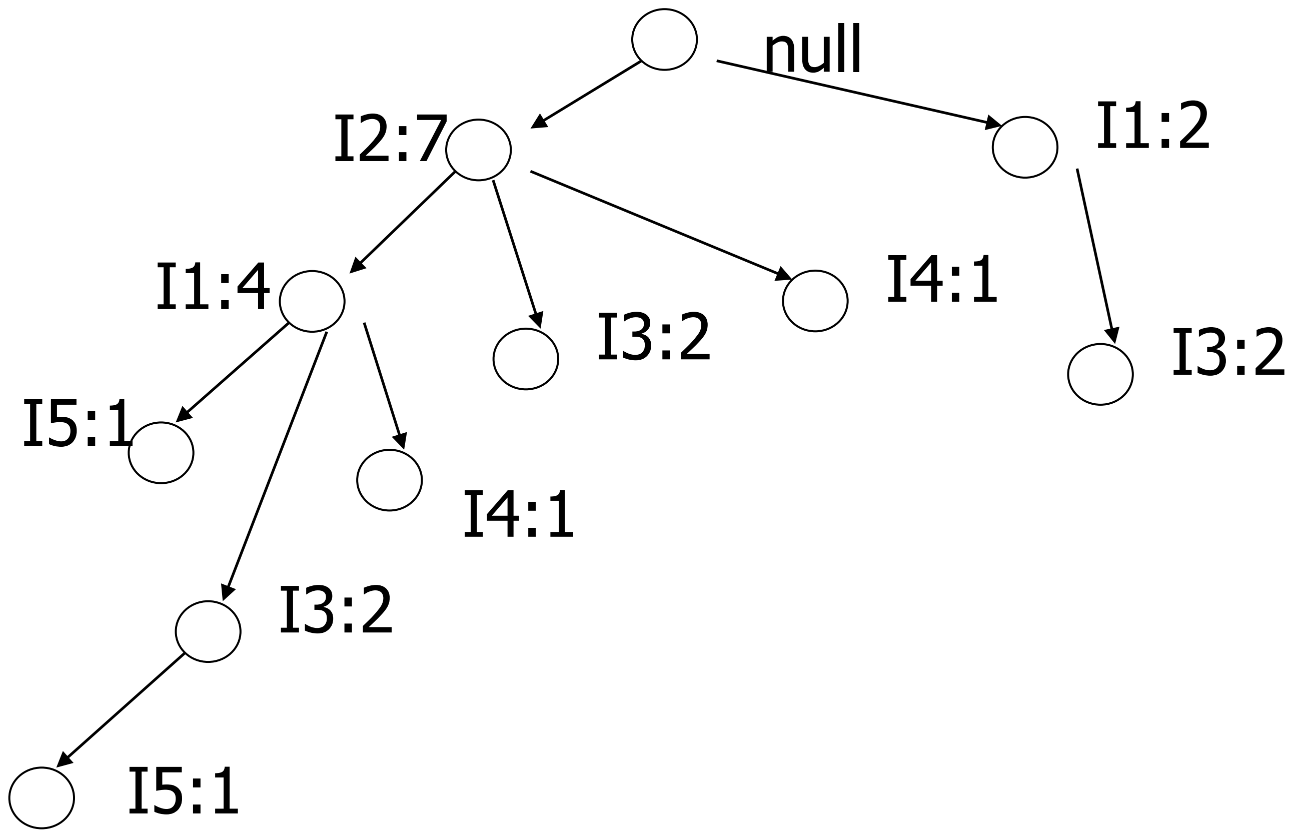
Order the items and create below branch for transaction T400: {I2,I1,I4}



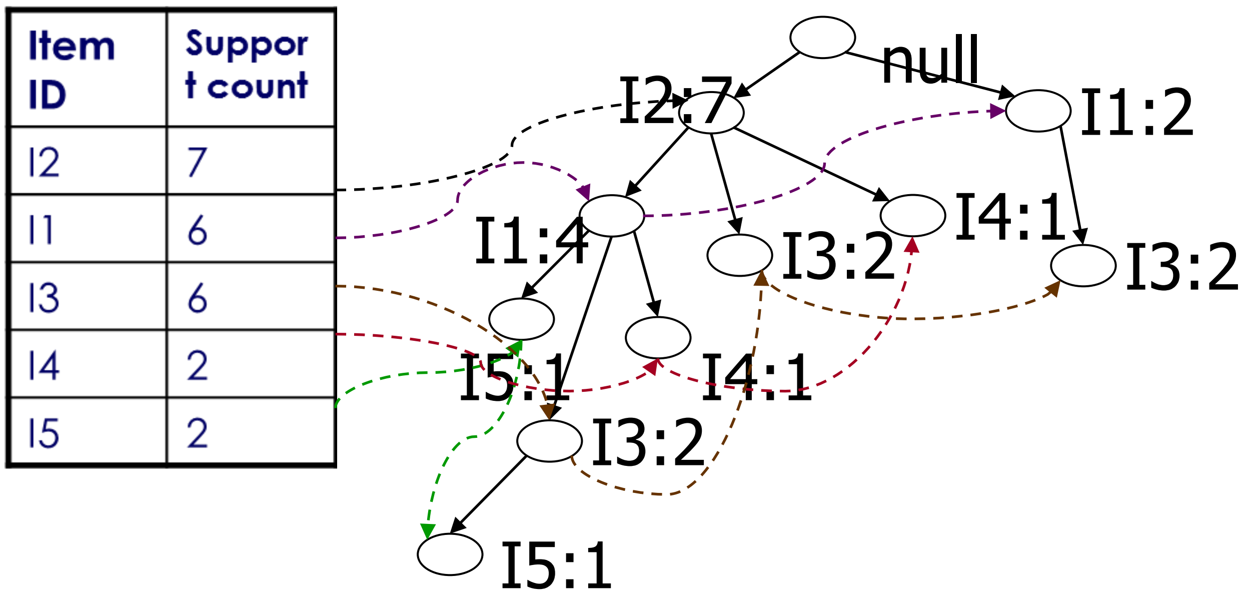
Order the items and create below branch for transaction T500: {I1,I3}



Order the items and create below branches for all remaining transactions



Final constructed tree with mapping to transaction table



Item ID	Support count
I2	7
I1	6
I3	6
I4	2
I5	2

The problem of mining frequent patterns in databases is transformed to that of mining the FP-tree

Create frequent patterns consisting I5

Occurrences of I5: <I2,I1,I5> and <I2,I1,I3,I5>

Two prefix Paths <I2, I1: 1> and <I2,I1,I3: 1>

Conditional FP tree contains only <I2: 2, I1: 2>, I3 is not considered because its support count of 1 is less than the minimum support count.

Frequent patterns {I2,I5:2}, {I1,I5:2},{I2,I1,I5:2}

<u>Create Frequent Patters for whole database</u>

TID	**Conditional Pattern Base**	**Conditional FP-tree**
I5	{{I2,I1:1},{I2,I1,I3:1}}	<I2:2,I1:2>
I4	{{I2,I1:1},{I2,1}}	<I2:2>
I3	{{I2,I1:2},{I2:2}, {I1:2}}	<I2:4,I1:2>,<I1:2>
I1	{I2,4}	<I2:4>

Table 3.4 FP Tree Conditional Pattern Base

⇩

TID	**Conditional FP-tree**	**Frequent Patterns Generated**
I5	<I2:2,I1:2>	{I2,I5:2}, {I1,I5:2},{I2,I1,I5:2}
I4	<I2:2>	{I2,I4:2}
I3	<I2:4,I1:2>,<I1:2>	{I2,I3:4},{I1,I3:4},{I2,I1,I3:2}
I1	<I2:4>	{I2,I1:4}

Table 3.5 FP Tree Frequent Patterns

Chapter 4

4. ALGORITHM OVERVIEW

4.1 Introduction

Proposed new algorithm is based on fact that numbers can be processed faster than strings. But often our most of data is in String form like if we talk about grocery store then name of objects will be name of vegetables, fruits, or food items. But in new proposed algorithm we will first converts these names of objects into numbers as per certain rule. These numbers are called magic number. Due to these particular magic numbers, this algorithm is called magic number algorithm. Once all objects or data items are represented by those numbers (or magic numbers) then this algorithm is applied and patterns can be mined in faster manner than the existing algorithms. Below is pseudo code for algorithm. Before pseudo code, below is description of various notation used.

TL	Transaction List: List of transaction where value of each item indicates sum magic number of items purchased in that transaction
MS	Magic Sum: Sum of magic number of all items purchased in a transaction
MTL	Magic Transaction List: Each item index in list represents magic sum and each item value indicates count of transactions having that magic sum

Table 4.1 Magic Number Notation

Assign numeric value to each item such that sum of numeric values of 2 sets will be equal only if purchased items in both transactions are same. Let’s call these numeric values a magic numbers.

Calculate sum of magic numbers of all transactions and prepare TL.

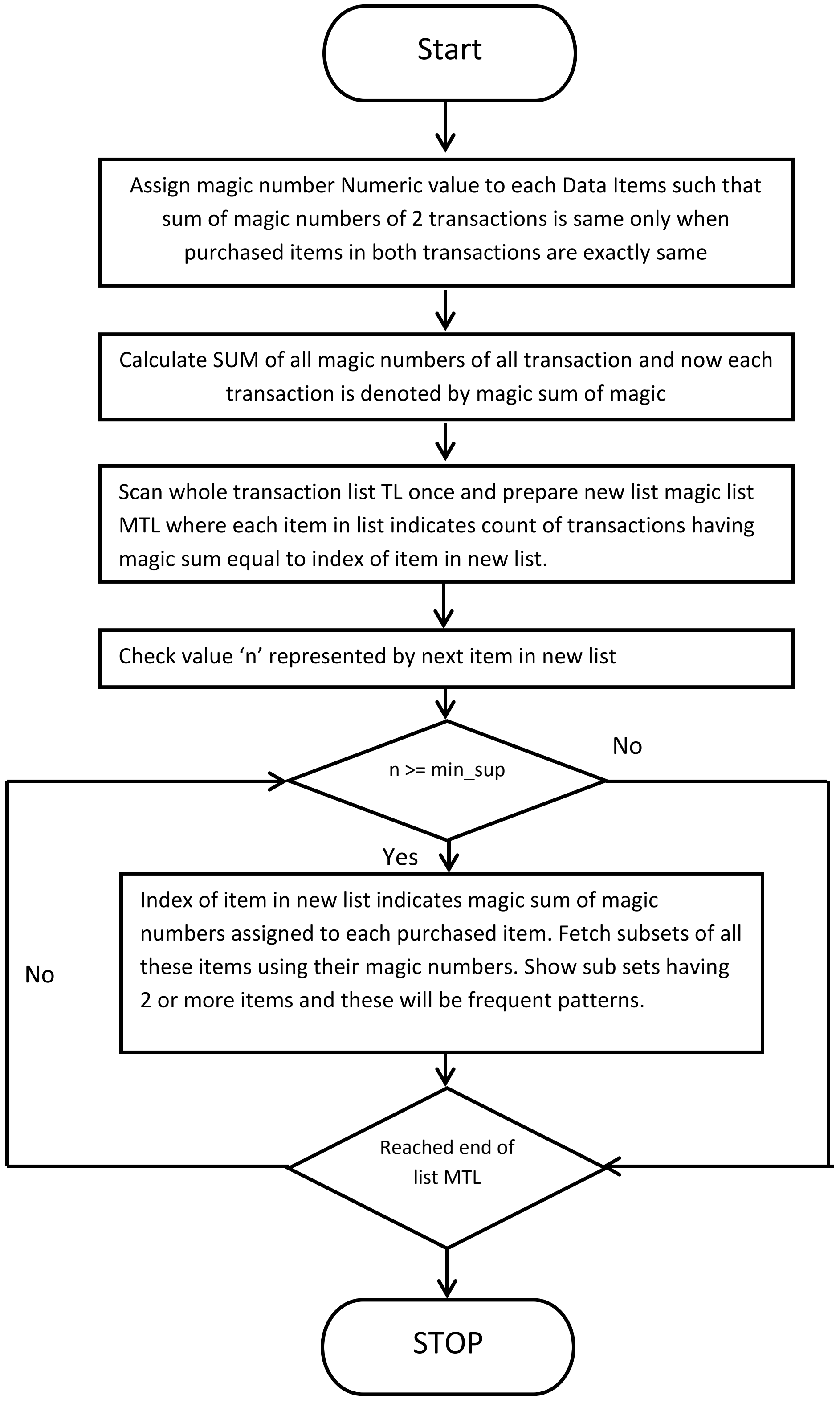
4.2 Pseudo Code

```
magic_number(TL)
{
    MTL = reduceDB(TL);   /* scan once and reduce the data base */
    for each item in MTL
        if ($MTL_{[count]}$ >= min_support)
        {
            Fetch subsets from Magic sum represented by ‘count’ and all subsets
            having 2 or more members are frequent patterns.
        }
}


List reduceDB(TL)
{
    for each item in TL with magic sum MS
    {
        $MTL_{[MS]} = MTL_{[MS]} + 1$;
        for each subset MSS of MS
            $MTL_{[MSS]} = MTL_{[MSS]} + 1$;
    }
    return MTL;
}
```

Answer: All subsets generated by magic_number();

4.3 Flow Diagram

Figure 4.1 Flow Diagram



Start
Assign magic number Numeric value to each Data Items such that sum of magic numbers of 2 transactions is same only when purchased items in both transactions are exactly same
Calculate SUM of all magic numbers of all transaction and now each transaction is denoted by magic sum of magic
Scan whole transaction list TL once and prepare new list magic list MTL where each item in list indicates count of transactions having magic sum equal to index of item in new list.
Check value 'n' represented by next item in new list
n >= min_sup
No
Yes
Index of item in new list indicates magic sum of magic numbers assigned to each purchased item. Fetch subsets of all these items using their magic numbers. Show sub sets having 2 or more items and these will be frequent patterns.
No
Reached end of list MTL
STOP

4.4 Algorithm Illustration With Example

- Let's assume a retail store chain like Wal-Mart , Big Bazar, Home-Plus etc, where they have various products like daily need things , cosmetic products , Stationary, Vegetables & Fruits.

- Owner company of these stores target is to mine the frequent patterns to know user preference and increase sales in future

- Below are 5 Transactions made at stationary section in a Retail store.

- Let's apply Magic Number Algorithm to mine frequent patterns on below transaction table. Min support level for FP= 3

T1	Eraser	Notebook	Pencil	Stapler
T2	Stapler	Pencil	Notebook	
T3	Eraser	Notebook	Sharpener	
T4	Notebook	Sharpener	Pencil	Stapler
T5	Pencil			

Table – 4.2 Magic Number Original Transaction Table

- After Applying Magic Number Algorithm, Table with occurrence count will become like as follow.

#	Eraser	Notebook	Pencil	Sharpener	Stapler	MAGIC SUM
T1	1	2	4	0	16	23
T2	0	2	4	0	16	22
T3	1	2	0	8	0	11
T4	0	2	4	8	16	30
T5	0	0	4	0	0	4
Occurrence Frequency	2	4	4	2	3	

Table – 4.3 Transaction Table with Magic Sum

- Each Item is represented by its assigned magic number.
- Magic Sum of each transaction is shown in last column

- After applying magic number new Magic transaction list will be as below

Transaction	Count	Transaction	Count
$T_{[0]}$	0	$T_{[17]}$	1
$T_{[1]}$	2	$T_{[18]}$	3
$T_{[2]}$	4	$T_{[19]}$	2
$T_{[3]}$	2	$T_{[20]}$	3
$T_{[4]}$	4	$T_{[21]}$	1
$T_{[5]}$	1	$T_{[22]}$	3
$T_{[6]}$	3	$T_{[23]}$	1
$T_{[7]}$	1	$T_{[24]}$	1
$T_{[8]}$	2	$T_{[25]}$	0
$T_{[9]}$	1	$T_{[26]}$	1
$T_{[10]}$	2	$T_{[27]}$	0
$T_{[11]}$	1	$T_{[28]}$	1
$T_{[12]}$	1	$T_{[29]}$	0
$T_{[13]}$	0	$T_{[30]}$	1
$T_{[14]}$	1	$T_{[31]}$	0
$T_{[15]}$	0		
$T_{[16]}$	3		

Table – 4.4 Transaction Table after applying Magic Number

- Now we can mine frequent pattern using Magic Number algorithm as follows.
- Magic sum that have Occurrence more than min support 3 are 2,4,6,16,18,20,22
- 2,4,16 are single items so not pair so we discard these numbers
- 6,18,20,22 are composite numbers, so these are 4 Frequent Patterns
- 6 = <Notebook, Pencil>
- 18 = < Notebook,Stapler>
- 20 = <Pencil, Stapler>
- 22 = <Notebook, Pencil,Stapler>

Chapter 5

5. ALGORITHM SIMULATION

Simulation of algorithm is done using Android application programming on Android Lollypop version 5.0. Implementation can be verified on any Android Mobile Device running on Android Lollypop version 5.0

5.1 Development Environment

OS	Android 5.0 (Lollypop)
SDK	ADT Build: v21.1.0-569685
CPU Capacity	ARM, 1.9GHz Quad Core
RAM	3 GB
API Level	17

Table 5.1 Development Environment

5.2 Software Program Structure

Simulation of algorithm is done using Android application programming on Android Lollipop version 5.0. Implementation can be verified on any Android Mobile Device running on Android Lollipop version 5.0

- Apriori algorithm Android Implementation majorly has 3 parts
  - src/MainActivity.java
  - res/activity_main.xml
  - AndroidManifest.xml

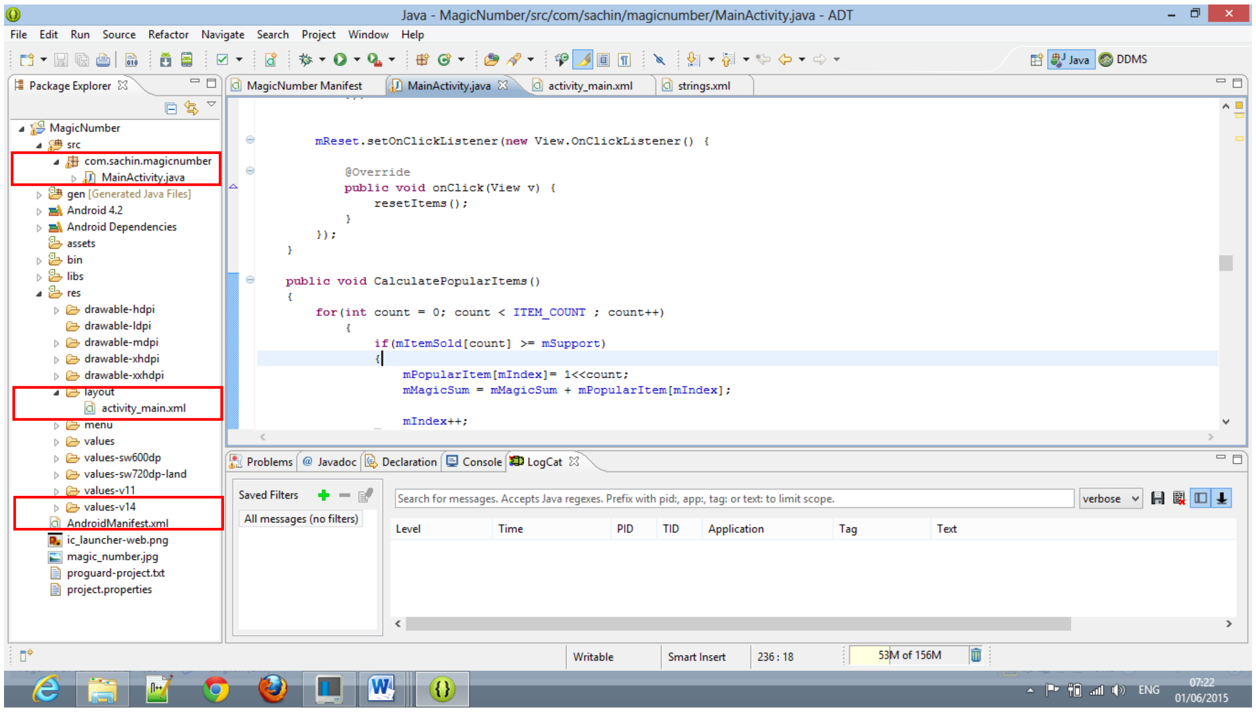
- **MainActivity.java:** This is main activity for implementation defines all data structures and data population in them, loading of GUI in application layout, logic related with mining of frequent patterns, defines all callback action on user interaction on various GUI components such as button etc.

- **Activity_main.xml:** This XML file defines all layouts, buttons, controls used in application like checkbox, buttons, textbox and their relative positioning, their color, paddings, text size etc

- **AndroidManifest.xml:** This XML define the meta data of applications like its version, Its name, Representing icon, priority, permission required, security parameters etc. it defines the launch mode in which application should be launched. Also application name and application icon are also defined here.

5.3 IDE Overview

Eclipse Integrated Development Environment is used for Software Development Purpose using Android SDK plugin in IDE.



```
        mReset.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                resetItems();
            }
        });
    }

    public void CalculatePopularItems()
    {
        for(int count = 0; count < ITEM_COUNT ; count++)
            {
                if(mItemSold[count] >= mSupport)
                {
                    mPopularItem[mIndex]= 1<<count;
                    mMagicSum = mMagicSum + mPopularItem[mIndex];

                    mIndex++;
```

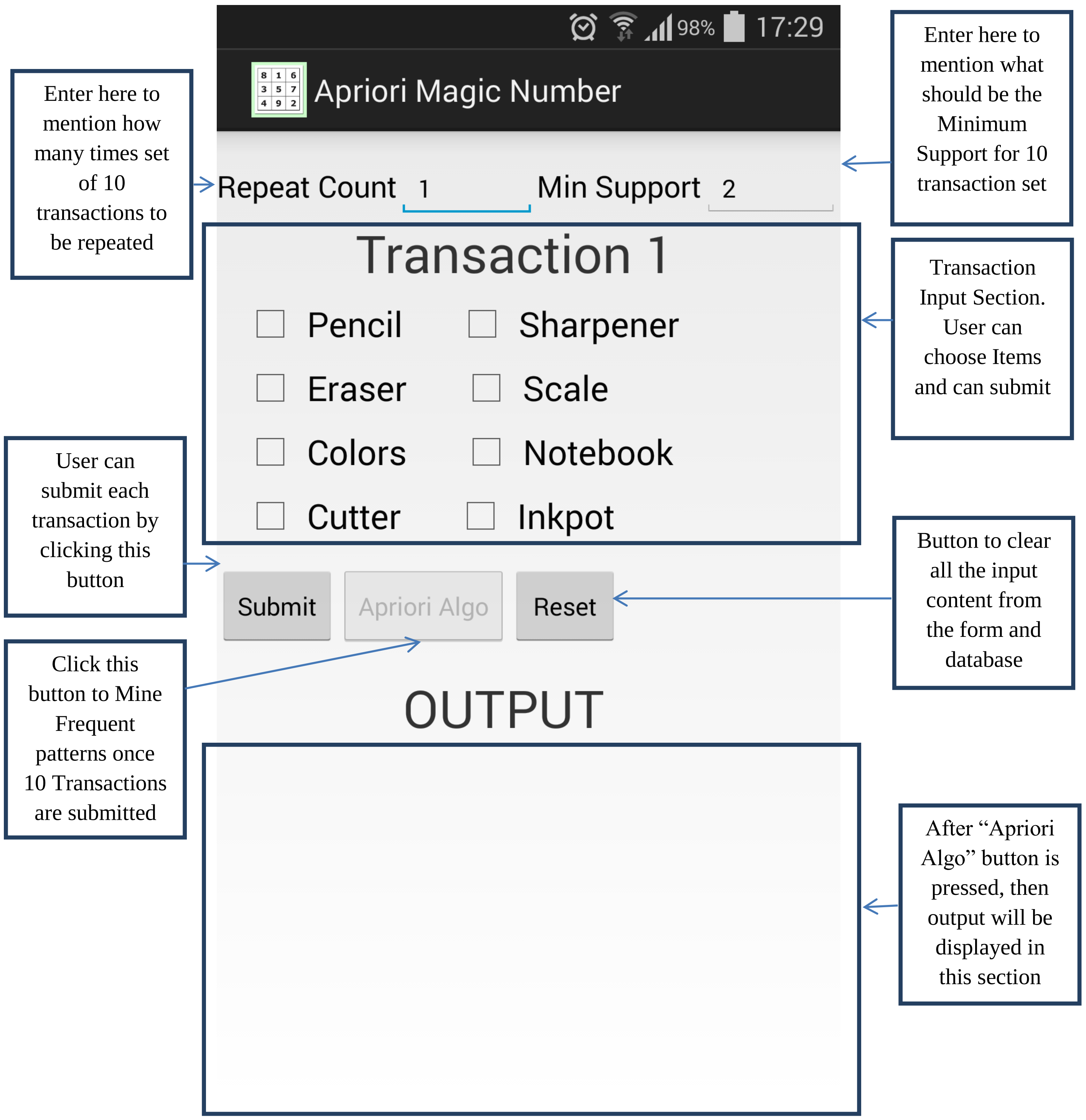
5.4 Input to Simulation Program

In Simulation program below was 10 transaction made.

Transaction #	Transaction Detail
T1	< Pencil, Eraser,Colors,Cutter>
T2	< Pencil, Eraser,Colors,Cutter>
T3	< Pencil, Eraser,Colors,Cutter>
T4	< Pencil, Eraser,Colors,Cutter>
T5	< Pencil, Eraser,Colors,Cutter>
T6	< Sharpener,Scale,Colors>
T7	<Sharpener,Notebook,Inkpot>
T8	<Colors,Cutter,Notebook>
T9	<Cutter,Inkpot>
T10	< Pencil, Cutter>

Table 5.2 Input Transaction Table

5.5 GUI Introduction



Enter here to mention how many times set of 10 transactions to be repeated

Enter here to mention what should be the Minimum Support for 10 transaction set

Transaction Input Section. User can choose Items and can submit

User can submit each transaction by clicking this button

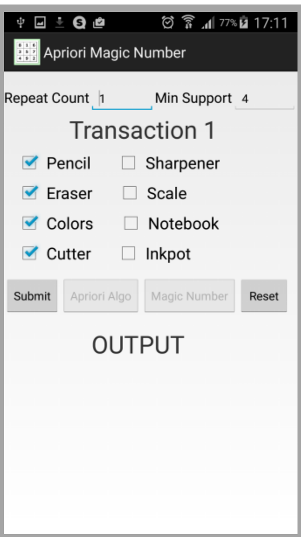
Button to clear all the input content from the form and database

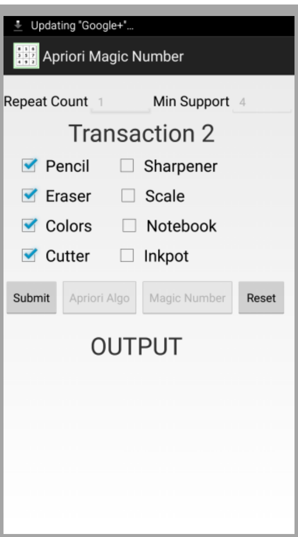
Click this button to Mine Frequent patterns once 10 Transactions are submitted

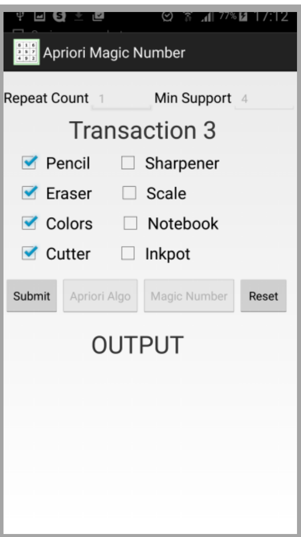
After “Apriori Algo” button is pressed, then output will be displayed in this section

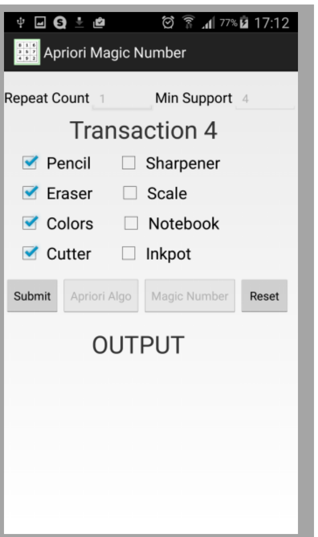
Figure 5.1 GUI Inroduction

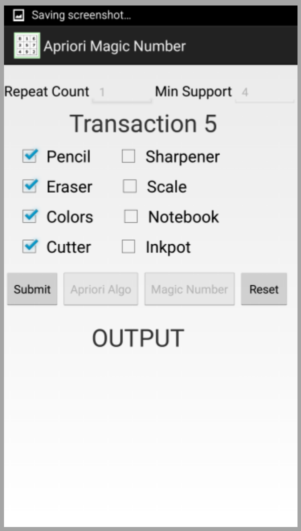
5.6 Input Screens Of Program

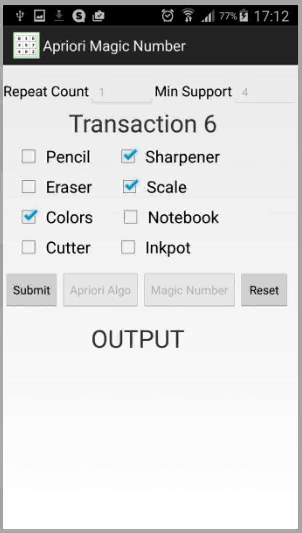

77% 17:11
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 1
✓ Pencil ☐ Sharpener
✓ Eraser ☐ Scale
✓ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

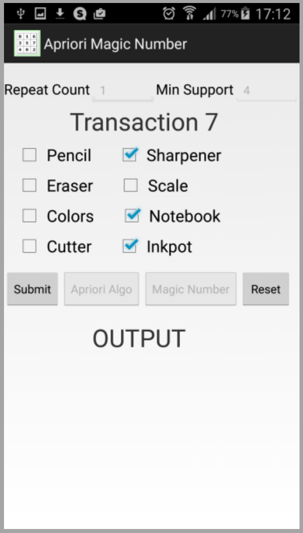

Updating "Google+"...
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 2
✓ Pencil ☐ Sharpener
✓ Eraser ☐ Scale
✓ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

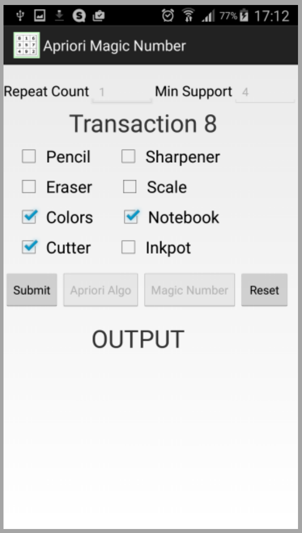

77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 3
✓ Pencil ☐ Sharpener
✓ Eraser ☐ Scale
✓ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

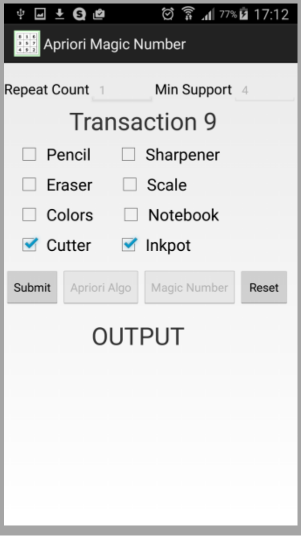

77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 4
✓ Pencil ☐ Sharpener
✓ Eraser ☐ Scale
✓ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

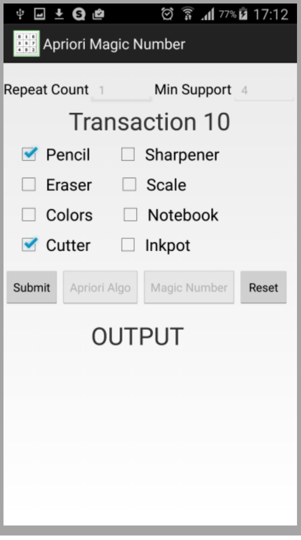

Saving screenshot...
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 5
✓ Pencil ☐ Sharpener
✓ Eraser ☐ Scale
✓ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT


77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 6
☐ Pencil ✓ Sharpener
☐ Eraser ✓ Scale
✓ Colors ☐ Notebook
☐ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT


77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 7
☐ Pencil ✓ Sharpener
☐ Eraser ☐ Scale
☐ Colors ✓ Notebook
☐ Cutter ✓ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

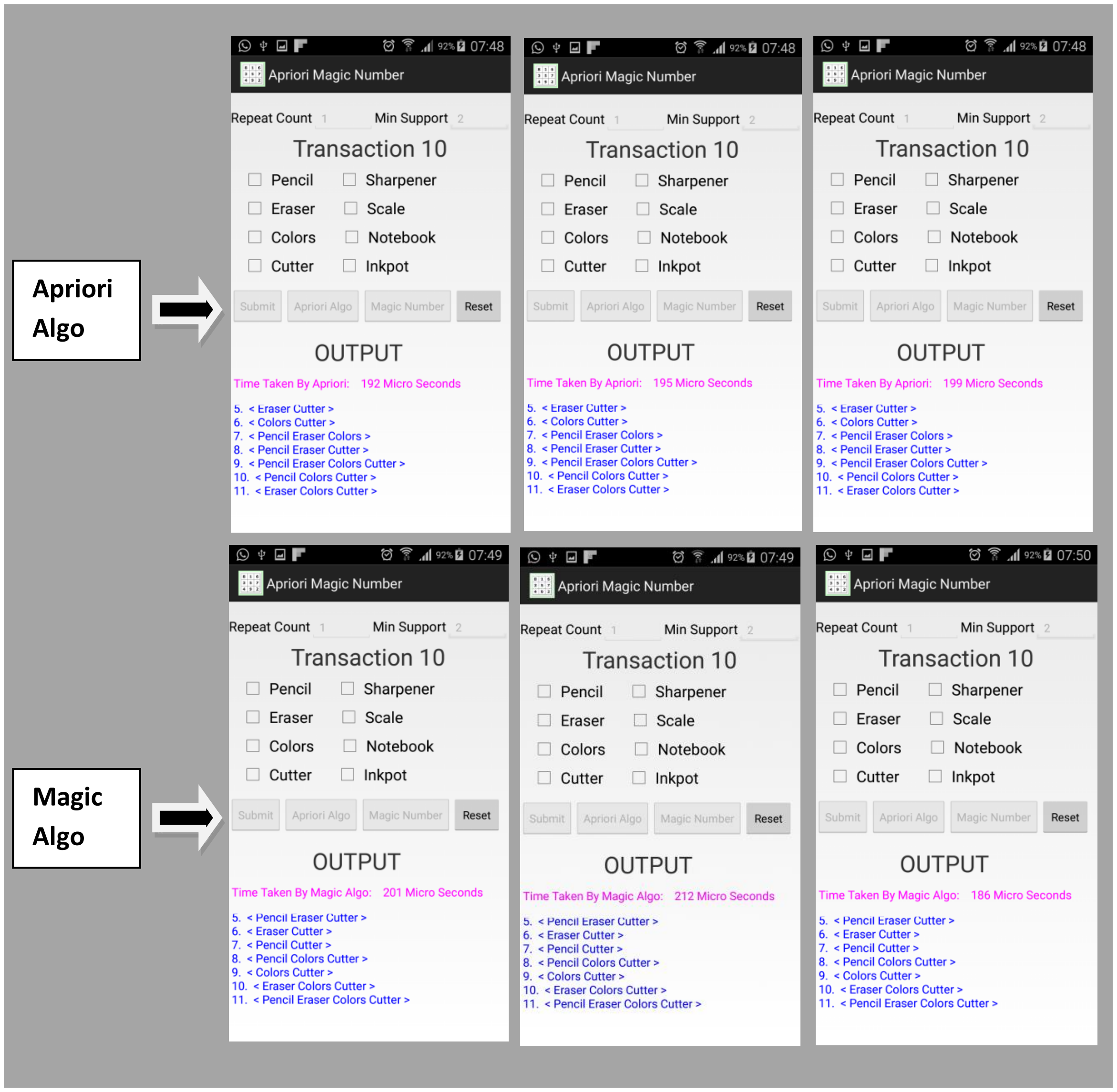

77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 8
☐ Pencil ☐ Sharpener
☐ Eraser ☐ Scale
✓ Colors ✓ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT


77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 9
☐ Pencil ☐ Sharpener
☐ Eraser ☐ Scale
☐ Colors ☐ Notebook
✓ Cutter ✓ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT


77% 17:12
Apriori Magic Number
Repeat Count 1 Min Support 4
Transaction 10
✓ Pencil ☐ Sharpener
☐ Eraser ☐ Scale
☐ Colors ☐ Notebook
✓ Cutter ☐ Inkpot
Submit | Apriori Algo | Magic Number | Reset
OUTPUT

5.7 Output From Program

min_support = 2, Transaction Count = 10



Repeat Count 1 Min Support 2

```
Time Taken By Apriori:   192 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   195 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   199 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   201 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

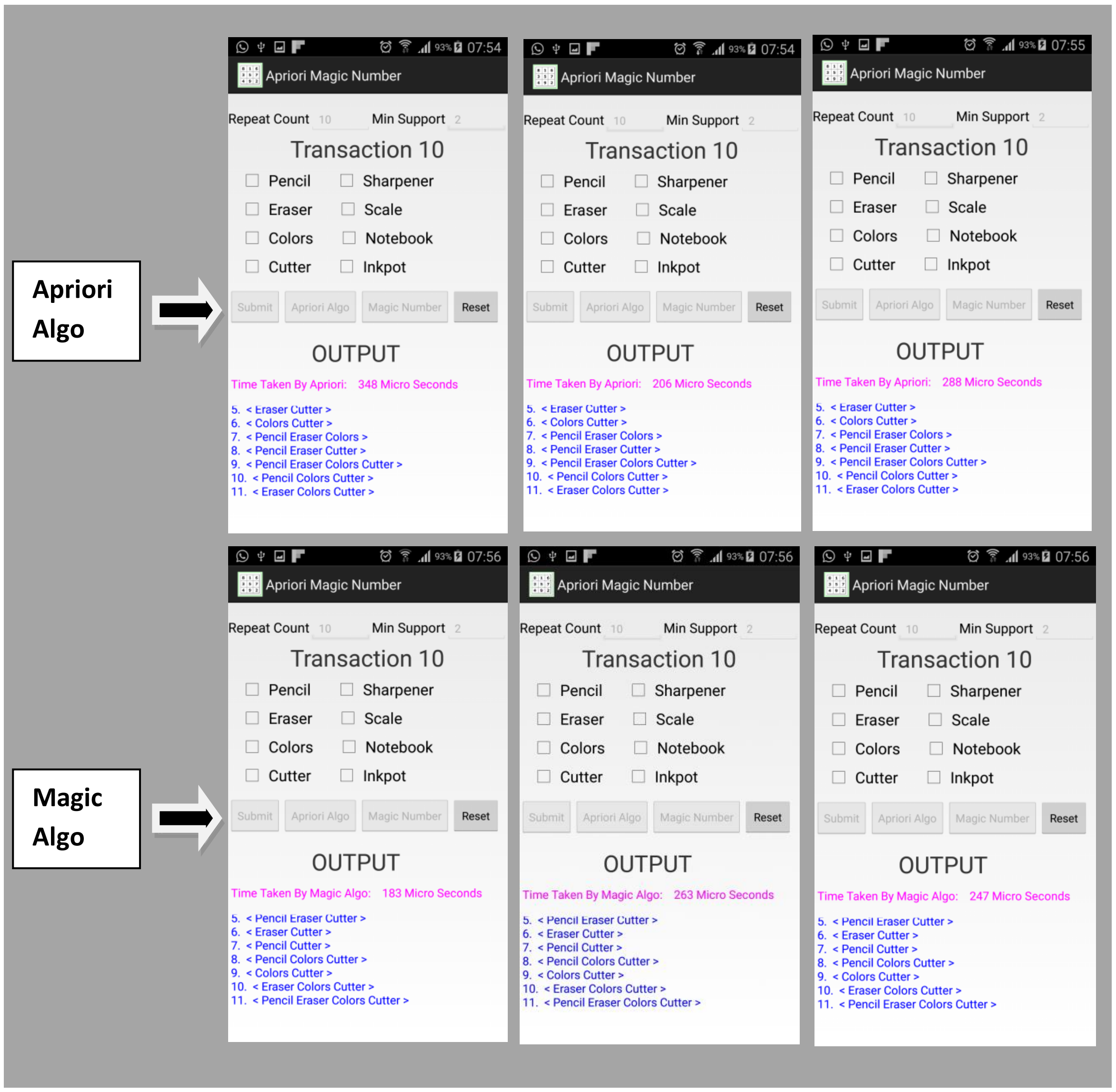
```
Time Taken By Magic Algo:   212 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   186 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (192+195+199)/3= **195 µSeconds**

Time taken by Magic Number Algorithm- (201+212+186)/3=**199 µSeconds**

min_support = 2, Transaction Count = 100



Repeat Count 10 Min Support 2

Transaction 10

Pencil, Sharpener, Eraser, Scale, Colors, Notebook, Cutter, Inkpot

OUTPUT

Time Taken By Apriori: 348 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10 Min Support 2

Transaction 10

OUTPUT

Time Taken By Apriori: 206 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10 Min Support 2

Transaction 10

OUTPUT

Time Taken By Apriori: 288 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10 Min Support 2

Transaction 10

OUTPUT

Time Taken By Magic Algo: 183 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Repeat Count 10 Min Support 2

Transaction 10

OUTPUT

Time Taken By Magic Algo: 263 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Repeat Count 10 Min Support 2

Transaction 10

OUTPUT

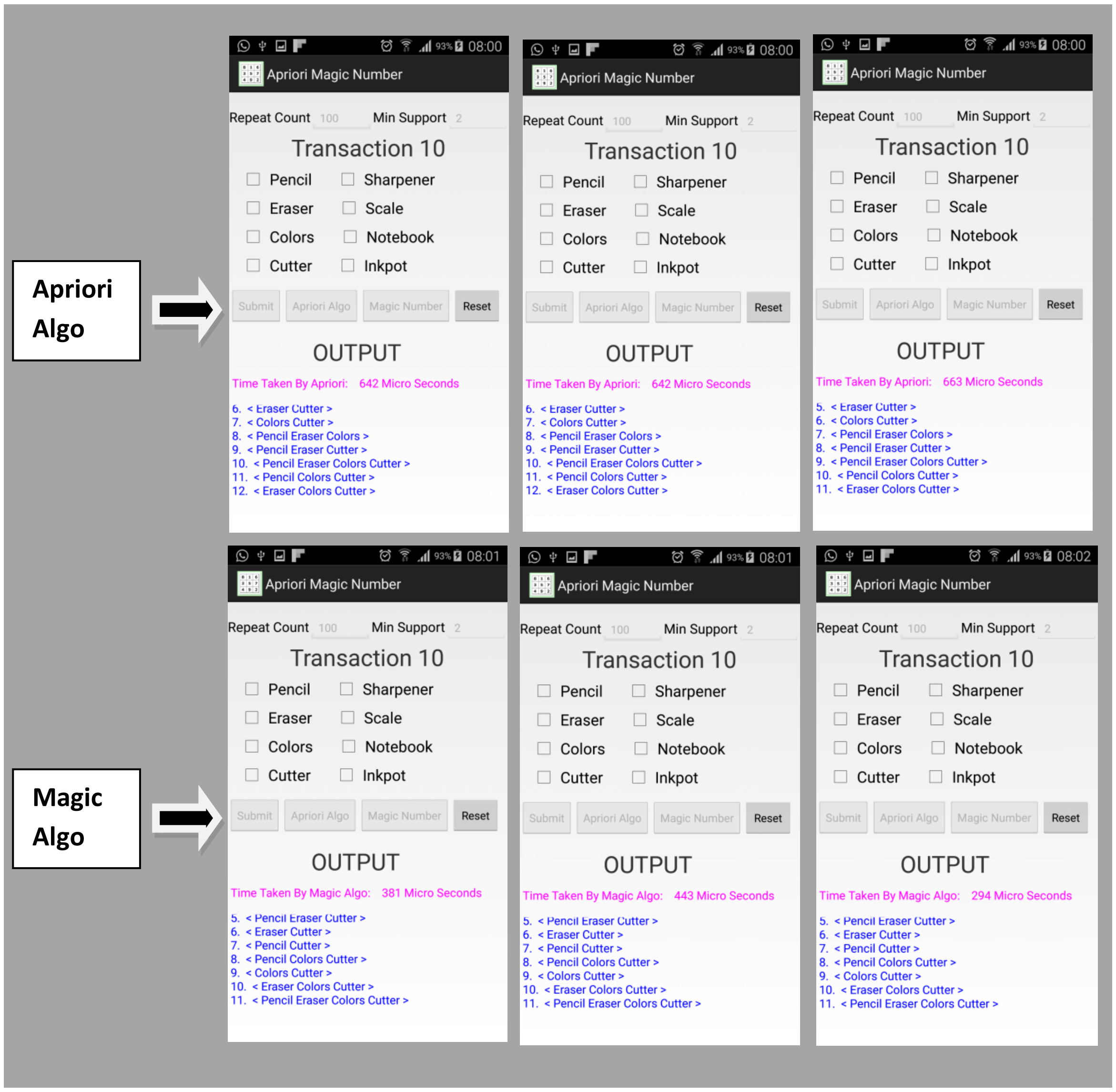
Time Taken By Magic Algo: 247 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Time taken by Apriori Algorithm- (348+206+288)/3= **280 µSeconds**

Time taken by Magic Number Algorithm- (183+263+247)/3=**231 µSeconds**

min_support = 2, Transaction Count = 1000



```
Time Taken By Apriori:   642 Micro Seconds
6.  < Eraser Cutter >
7.  < Colors Cutter >
8.  < Pencil Eraser Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser Colors Cutter >
11.  < Pencil Colors Cutter >
12.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   642 Micro Seconds
6.  < Eraser Cutter >
7.  < Colors Cutter >
8.  < Pencil Eraser Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser Colors Cutter >
11.  < Pencil Colors Cutter >
12.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   663 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   381 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

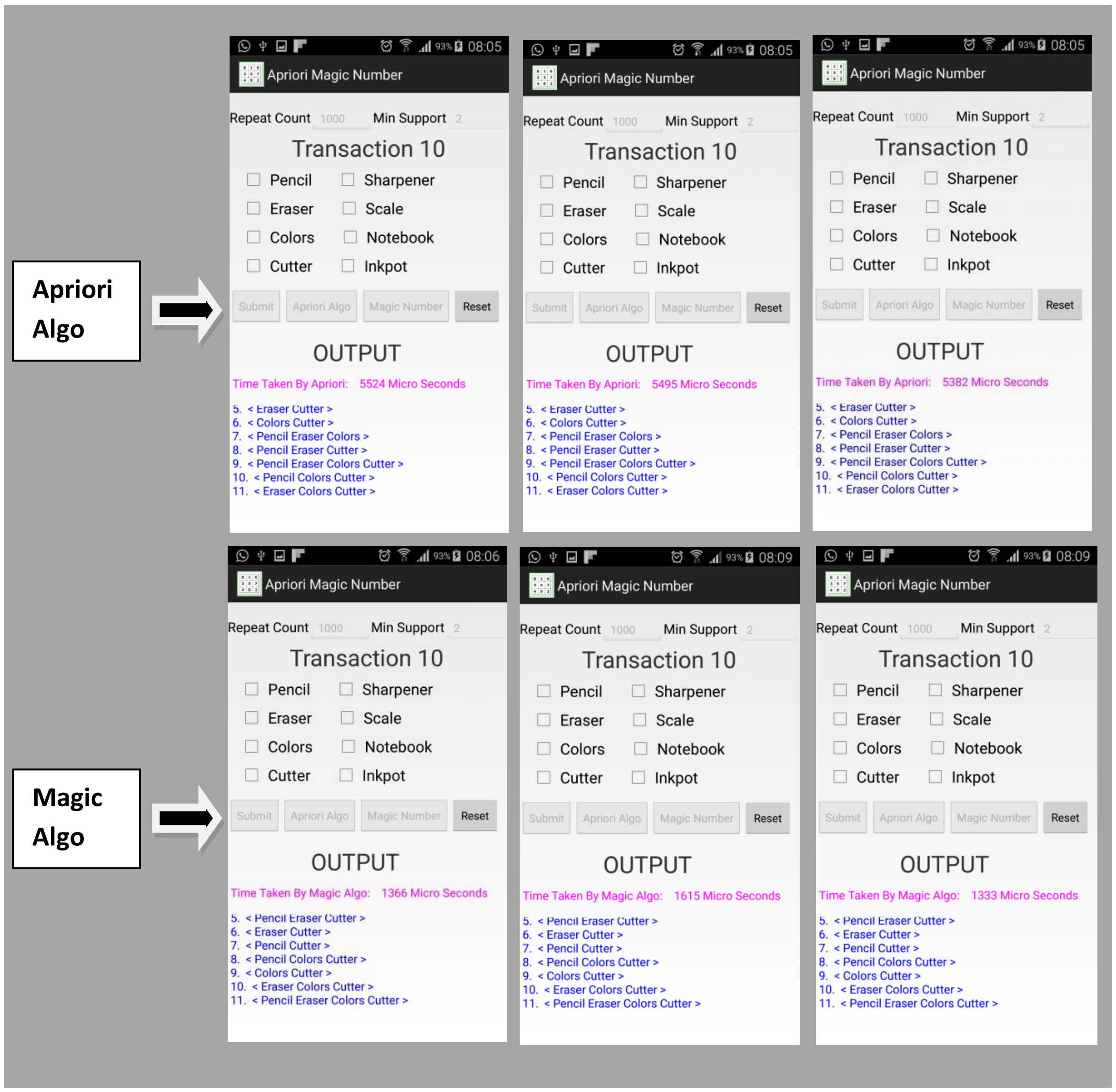
```
Time Taken By Magic Algo:   443 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   294 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (642+642+6663)/3= **649 µSeconds**

Time taken by Magic Number Algorithm- (381+443+294)/3=**372 µSeconds**

min_support = 2, Transaction Count = 10,000



93% 08:05
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Apriori: 5524 Micro Seconds
5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

93% 08:05
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Apriori: 5495 Micro Seconds
5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

93% 08:05
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Apriori: 5382 Micro Seconds
5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

93% 08:06
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Magic Algo: 1366 Micro Seconds
5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

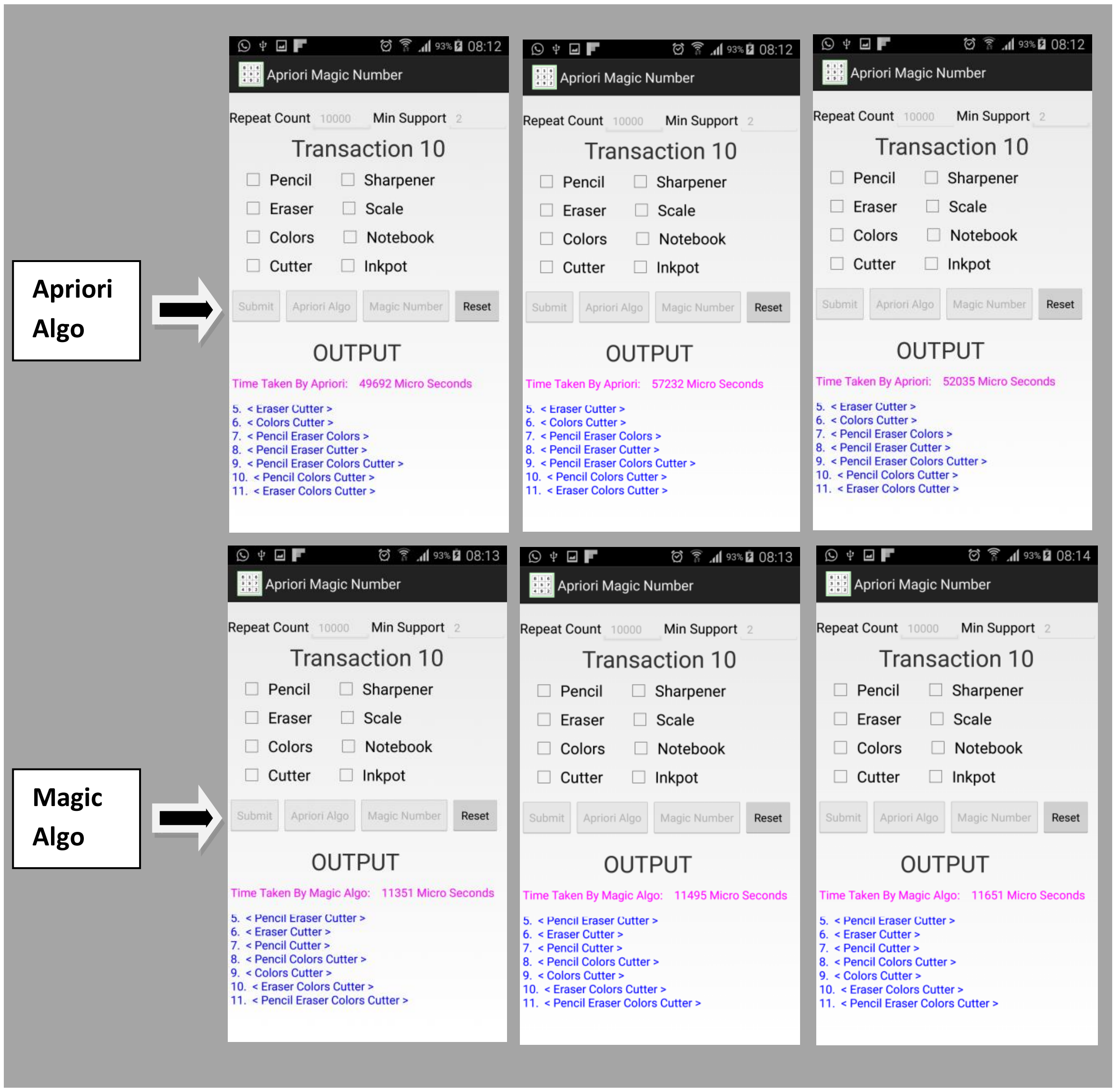
93% 08:09
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Magic Algo: 1615 Micro Seconds
5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

93% 08:09
Apriori Magic Number
Repeat Count 1000 Min Support 2
Transaction 10
Pencil Sharpener
Eraser Scale
Colors Notebook
Cutter Inkpot
Submit Apriori Algo Magic Number Reset
OUTPUT
Time Taken By Magic Algo: 1333 Micro Seconds
5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Time taken by Apriori Algorithm- (5542+5495+5382)/3= **5473 µSeconds**

Time taken by Magic Number Algorithm- (1366+1615+1333)/3=**1438 µSeconds**

min_support = 2, Transaction Count = 100,000



```
Repeat Count 10000   Min Support 2

Time Taken By Apriori:   49692 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 2

Time Taken By Apriori:   57232 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 2

Time Taken By Apriori:   52035 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 2

Time Taken By Magic Algo:   11351 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

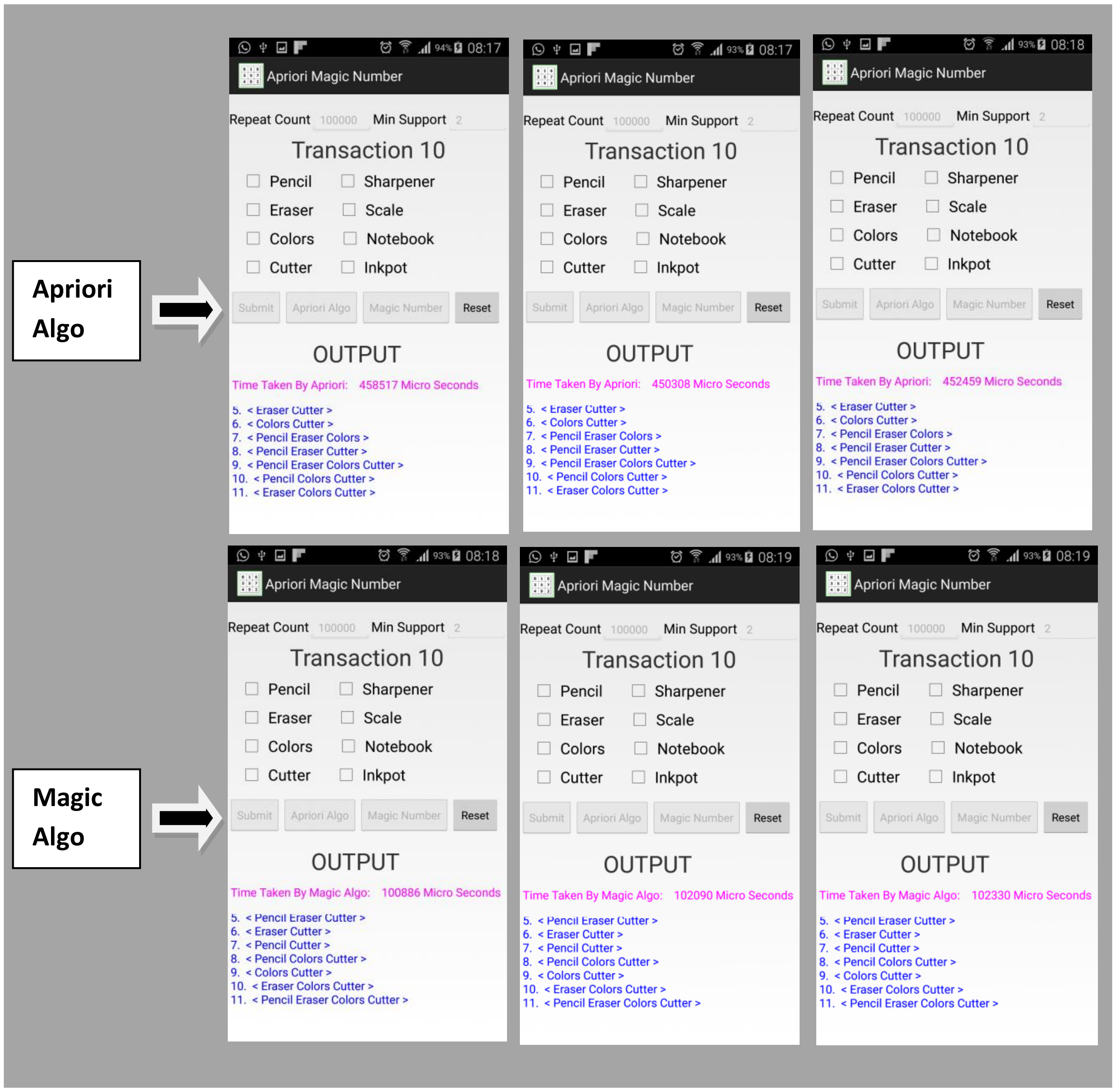
```
Repeat Count 10000   Min Support 2

Time Taken By Magic Algo:   11495 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 2

Time Taken By Magic Algo:   11651 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (49692+57232+52035)/3= **52986 µSeconds**

Time taken by Magic Number Algorithm- (11351+11495+11651)/3=**11499 µSeconds**

min_support = 2, Transaction Count = 1000,000



```
Repeat Count 100000    Min Support 2

Time Taken By Apriori:   458517 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000    Min Support 2

Time Taken By Apriori:   450308 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000    Min Support 2

Time Taken By Apriori:   452459 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000    Min Support 2

Time Taken By Magic Algo:   100886 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000    Min Support 2

Time Taken By Magic Algo:   102090 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000    Min Support 2

Time Taken By Magic Algo:   102330 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (454517+450308+452459)/3= **452428 µSeconds**

Time taken by Magic Number Algorithm- (100886+102090+102330)/3=**101768 µSeconds**

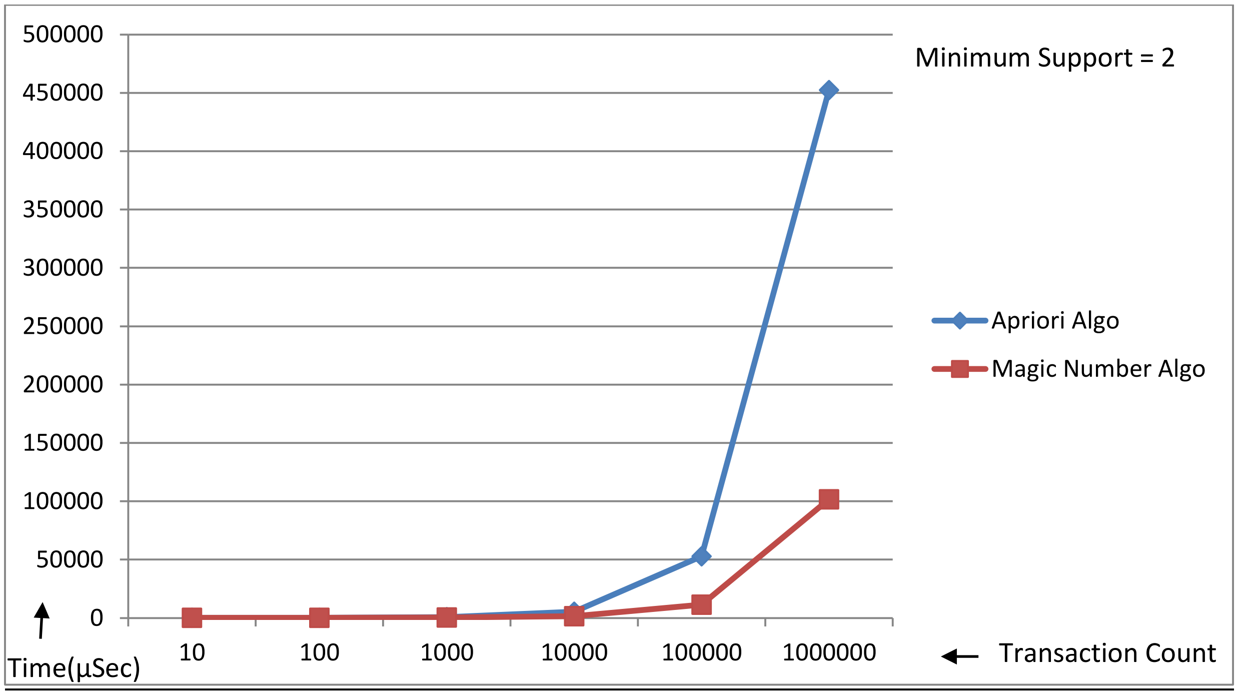
Execution Performance Comparison In Tabular Form

Min_support = 2

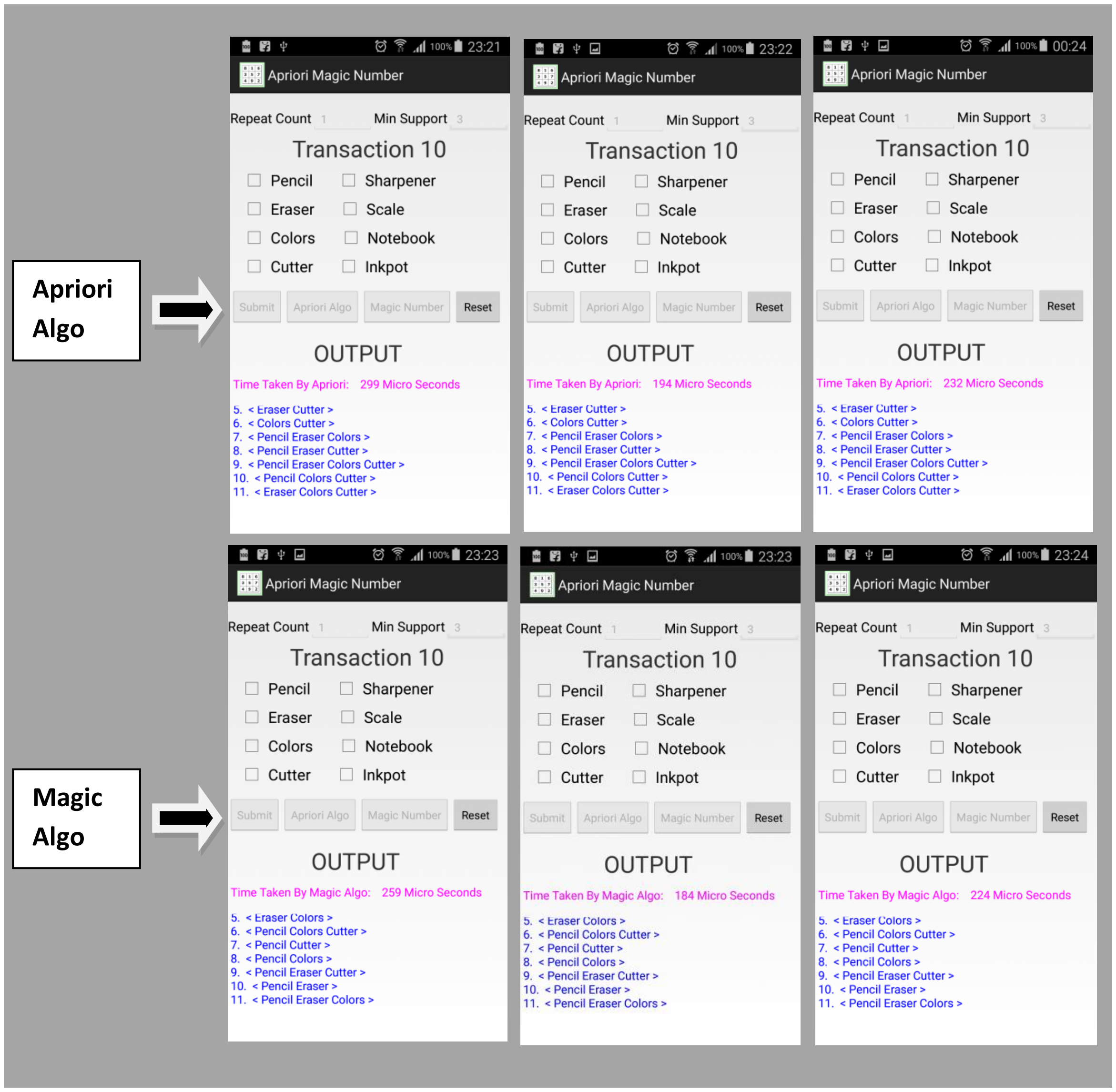
Transaction Count	**Time By Apriori(µSec)**	**Time by Magic Number(µSec)**
10	195	199
100	280	231
1000	649	372
10000	5473	1438
100000	52986	11496
1000000	452428	101768

Table 5.3 Output Table with Min Support 2

Execution Performance Comparision by Graph



min_support = 3, Transaction Count = 10



```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Apriori:   299 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Apriori:   194 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Apriori:   232 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Magic Algo:   259 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

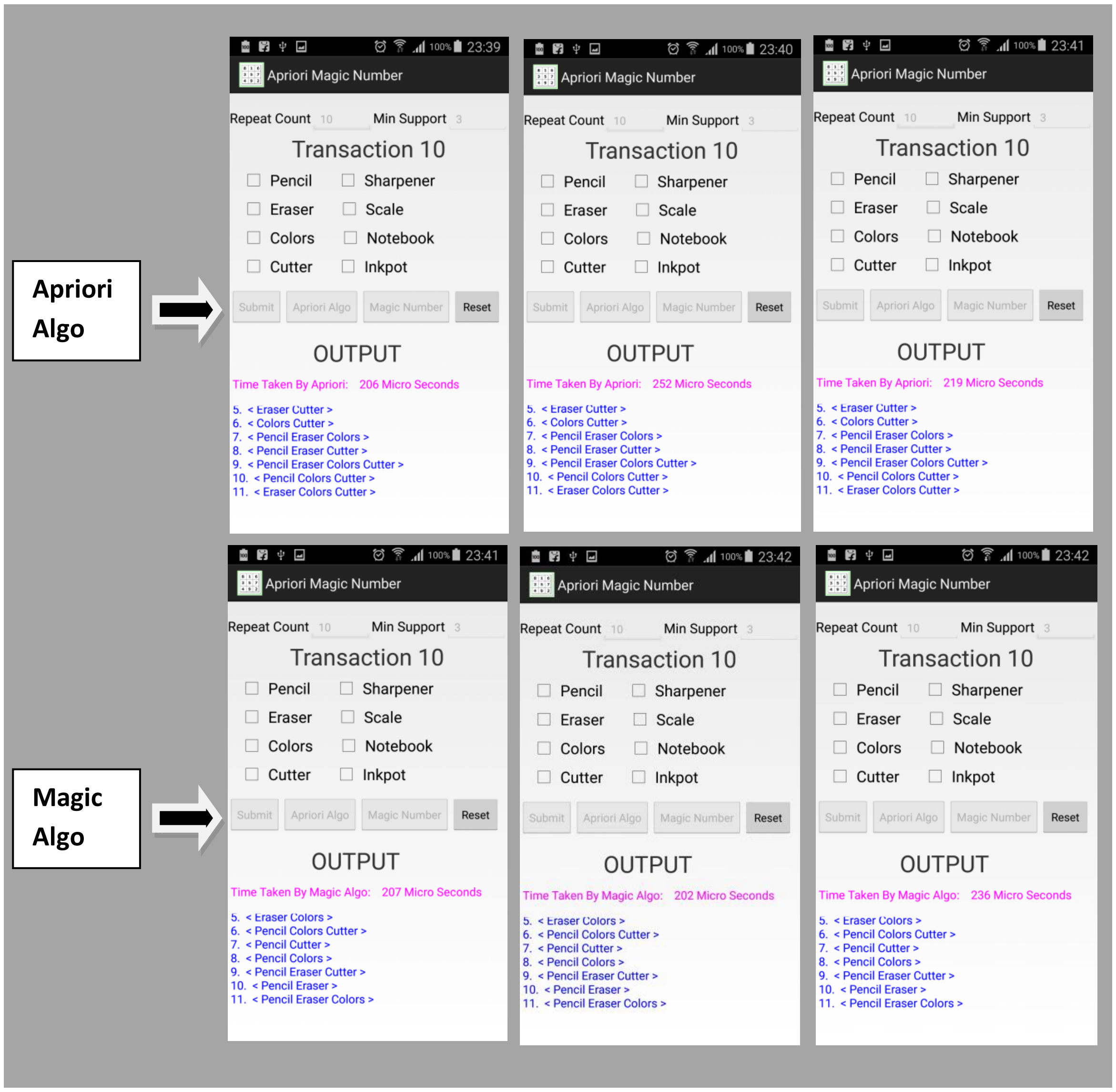
```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Magic Algo:   184 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

```
Repeat Count 1    Min Support 3
Transaction 10
OUTPUT
Time Taken By Magic Algo:   224 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (299+194+232)/3= **241 µSeconds**

Time taken by Magic Number Algorithm- (259+184+224)/3=**222 µSeconds**

min_support = 3, Transaction Count = 100



```
23:39
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Apriori:   206 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
23:40
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Apriori:   252 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
23:41
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Apriori:   219 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
23:41
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Magic Algo:   207 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

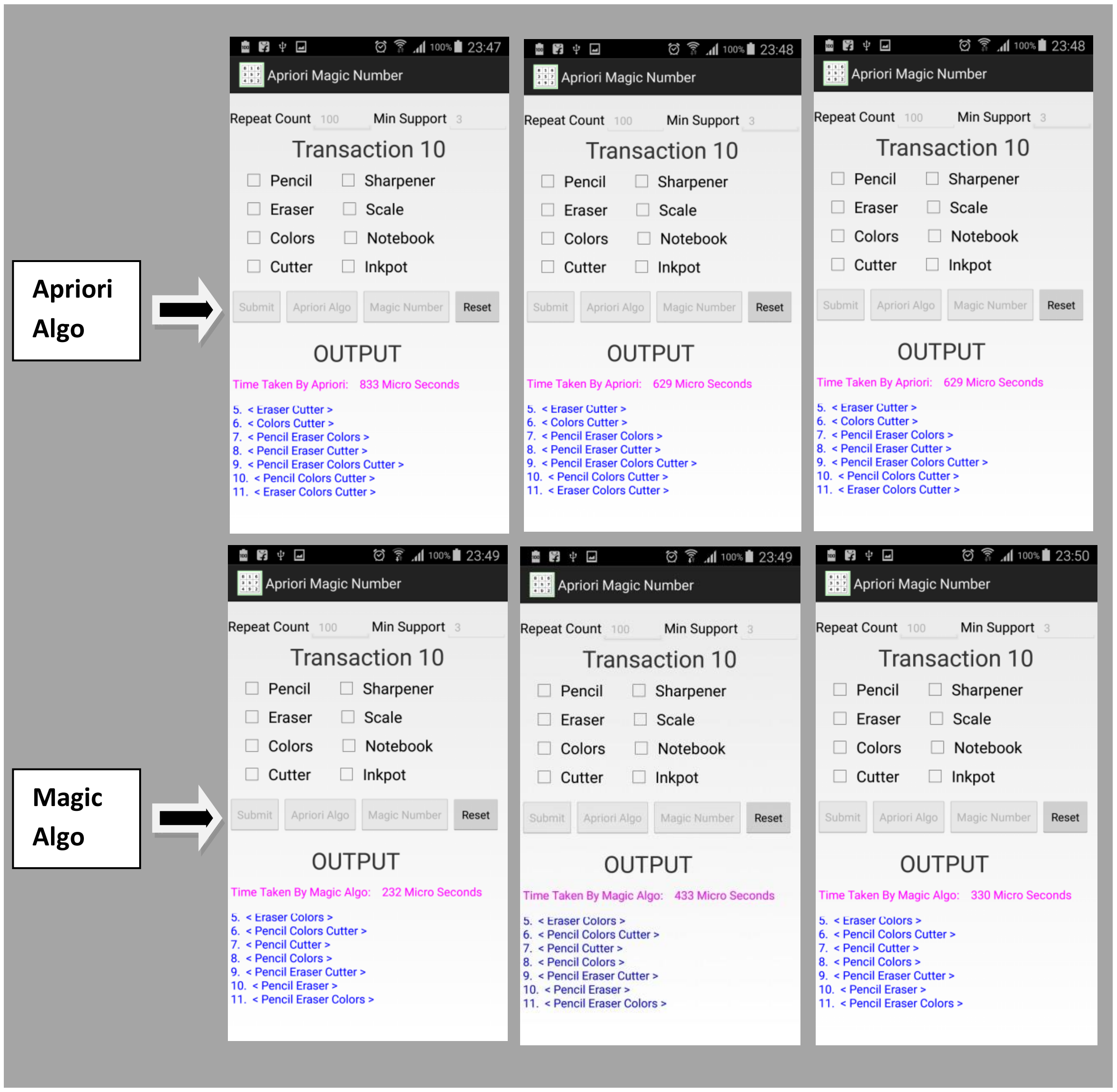
```
23:42
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Magic Algo:   202 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

```
23:42
Apriori Magic Number
Repeat Count 10    Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot
Submit  Apriori Algo  Magic Number  Reset
OUTPUT
Time Taken By Magic Algo:   236 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (206+252+219)/3= **225 µSeconds**

Time taken by Magic Number Algorithm- (207+202+236)/3=**215 µSeconds**

min_support = 3, Transaction Count = 1000



Repeat Count 100 Min Support 3

```
Time Taken By Apriori:   833 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   629 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   629 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   232 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

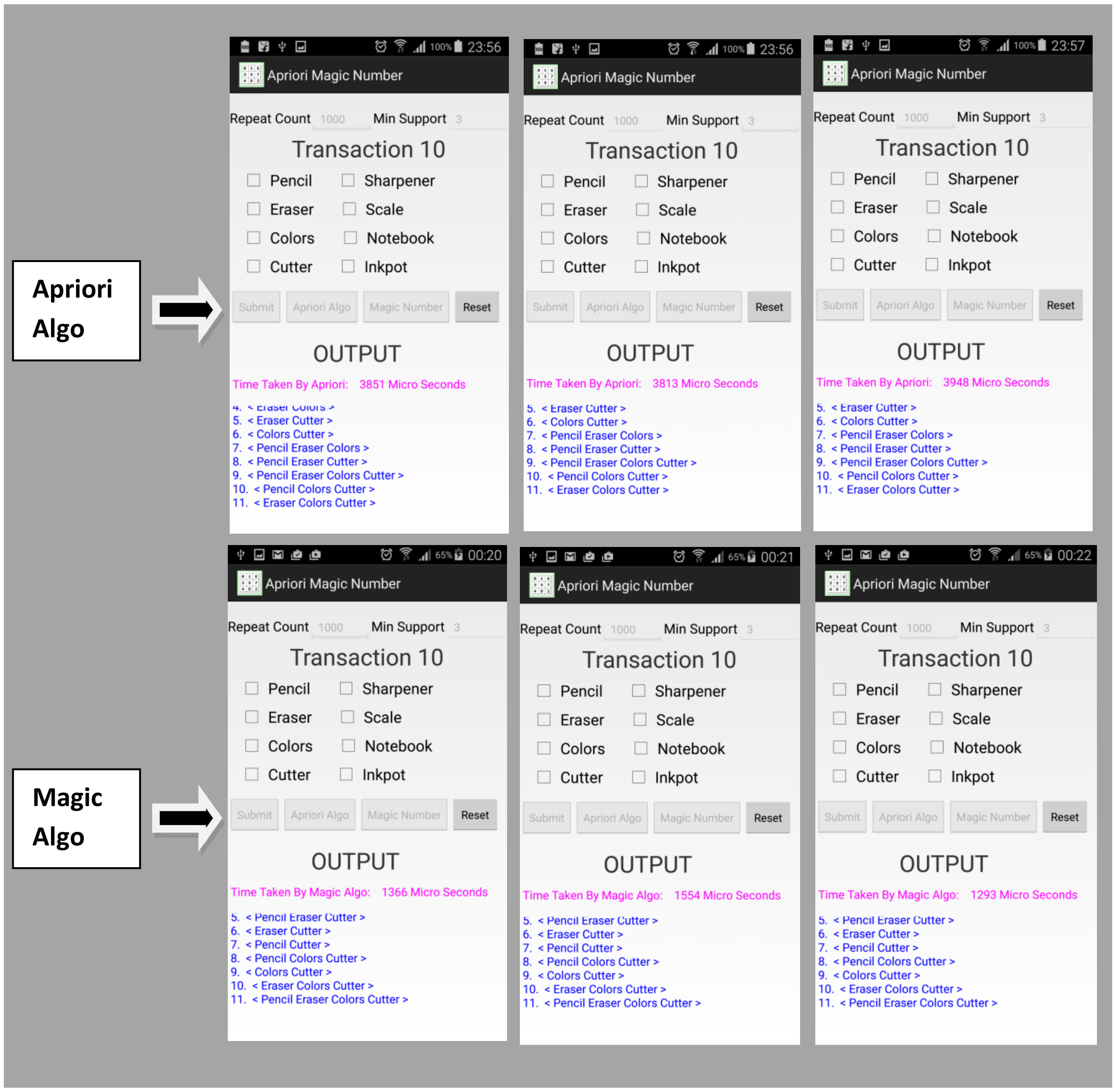
```
Time Taken By Magic Algo:   433 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

```
Time Taken By Magic Algo:   330 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (833+429+629)/3= **630 µSeconds**

Time taken by Magic Number Algorithm- (232+433+330)/3=**331 µSeconds**

min_support = 3, Transaction Count = 10,000



```
Time Taken By Apriori:   3851 Micro Seconds
4.  < Eraser Colors >
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   3813 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   3948 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1366 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

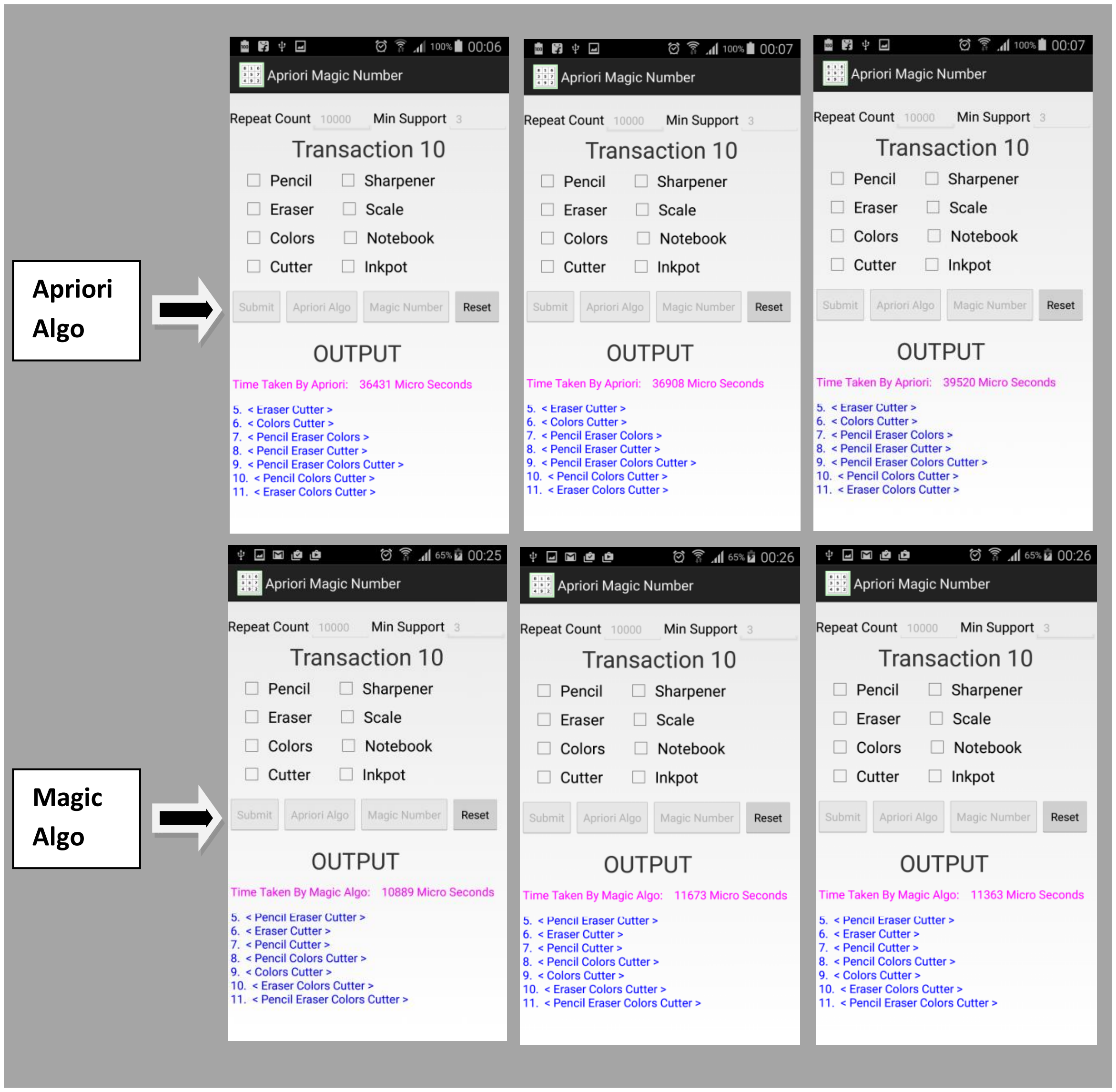
```
Time Taken By Magic Algo:   1554 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1293 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (3851+3813+3948)/3= **3870 µSeconds**

Time taken by Magic Number Algorithm- (1366+1554+1293)/3=**1404 µSeconds**

min_support = 3, Transaction Count = 100,000



Repeat Count 10000 Min Support 3

Time Taken By Apriori: 36431 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10000 Min Support 3

Time Taken By Apriori: 36908 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10000 Min Support 3

Time Taken By Apriori: 39520 Micro Seconds

5. < Eraser Cutter >
6. < Colors Cutter >
7. < Pencil Eraser Colors >
8. < Pencil Eraser Cutter >
9. < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >

Repeat Count 10000 Min Support 3

Time Taken By Magic Algo: 10889 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Repeat Count 10000 Min Support 3

Time Taken By Magic Algo: 11673 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Repeat Count 10000 Min Support 3

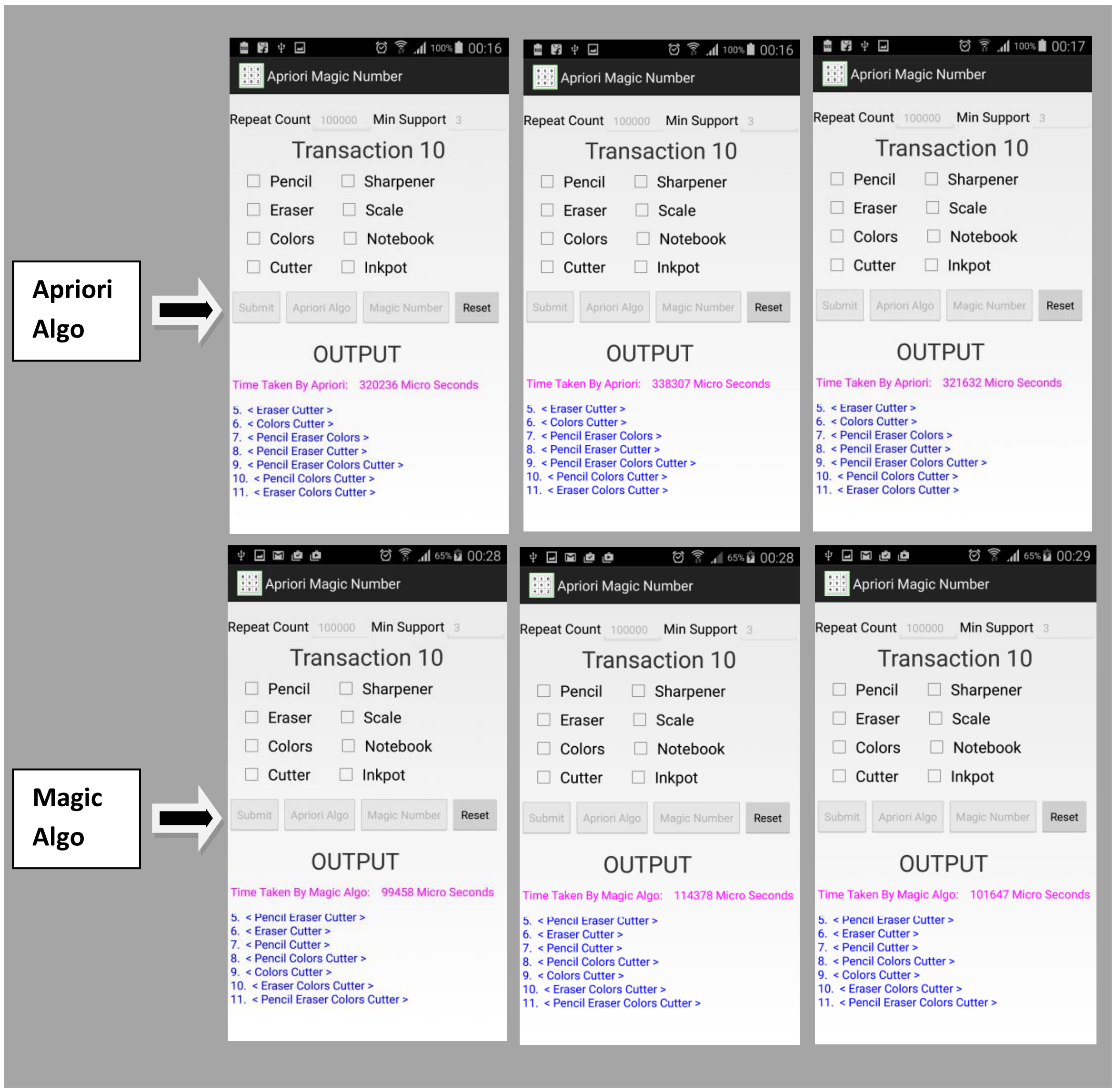
Time Taken By Magic Algo: 11363 Micro Seconds

5. < Pencil Eraser Cutter >
6. < Eraser Cutter >
7. < Pencil Cutter >
8. < Pencil Colors Cutter >
9. < Colors Cutter >
10. < Eraser Colors Cutter >
11. < Pencil Eraser Colors Cutter >

Time taken by Apriori Algorithm- (36431+36908+39520)/3= **37620 µSeconds**

Time taken by Magic Number Algorithm- (10889+11673+11363)/3=**11308 µSeconds**

min_support = 3, Transaction Count = 1000,000



```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Apriori:   320236 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Apriori:   338307 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Apriori:   321632 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Magic Algo:   99458 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Magic Algo:   114378 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 3
Transaction 10
Pencil   Sharpener
Eraser   Scale
Colors   Notebook
Cutter   Inkpot

Time Taken By Magic Algo:   101647 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (320036+338307+321632)/3= **326658 μSeconds**

Time taken by Magic Number Algorithm- (99458+114378+101647)/3=**105161 μSeconds**

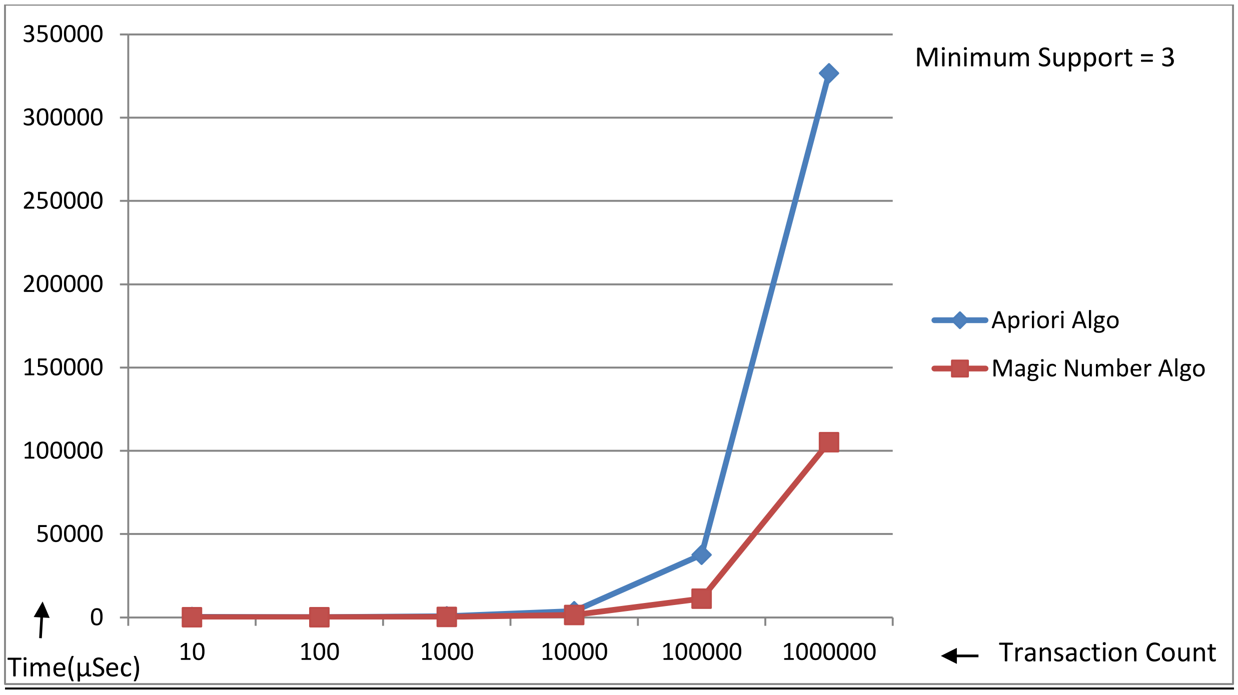
Execution Performance Comparison In Tabular Form

Min_support = 3

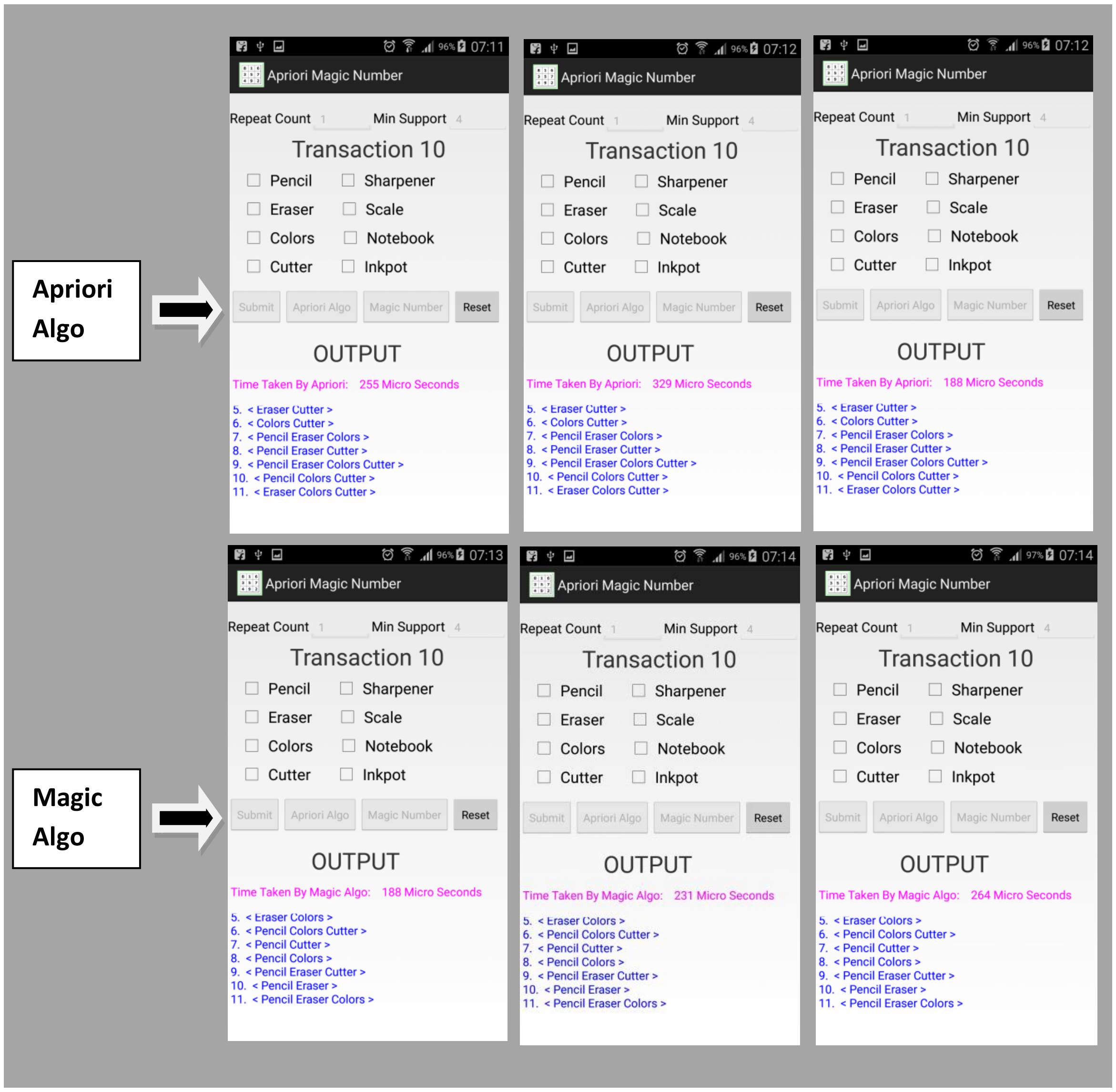
Transaction Count	Time By Apriori(μSec)	Time by Magic Number(μSec)
10	241	222
100	225	215
1000	630	331
10000	3870	1404
100000	37620	11308
1000000	326658	105161

Table 5.4 Output Table with Min Support 3

Execution Performance Comparision by Graph



min_support = 4, Transaction Count = 10



Apriori Algo, run 1 (07:11): Repeat Count 1, Min Support 4

```
Time Taken By Apriori:   255 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Apriori Algo, run 2 (07:12): Repeat Count 1, Min Support 4

```
Time Taken By Apriori:   329 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Apriori Algo, run 3 (07:12): Repeat Count 1, Min Support 4

```
Time Taken By Apriori:   188 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Magic Algo, run 1 (07:13): Repeat Count 1, Min Support 4

```
Time Taken By Magic Algo:   188 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Magic Algo, run 2 (07:14): Repeat Count 1, Min Support 4

```
Time Taken By Magic Algo:   231 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

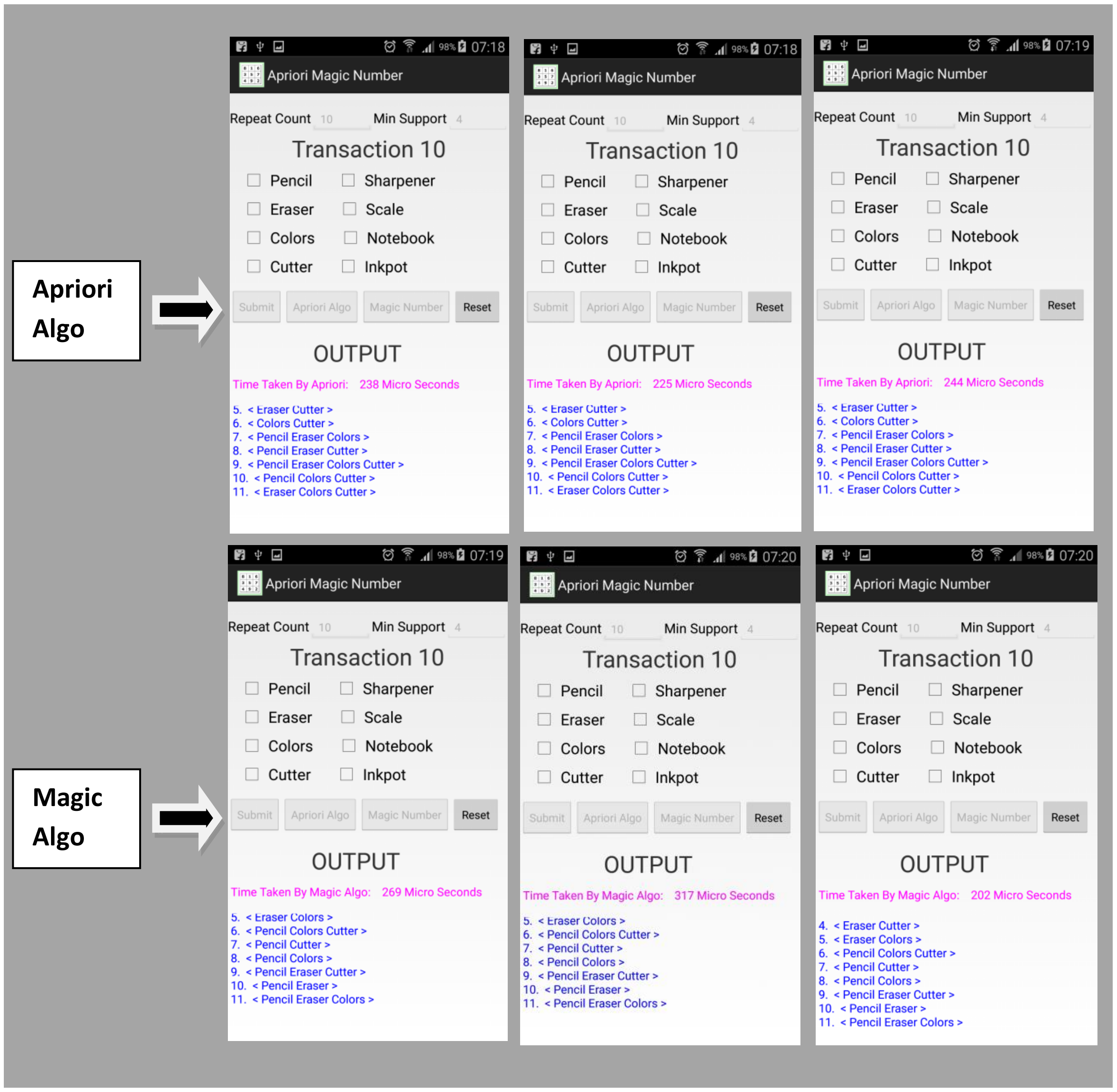
Magic Algo, run 3 (07:14): Repeat Count 1, Min Support 4

```
Time Taken By Magic Algo:   264 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (255+329+188)/3= **257 µSeconds**

Time taken by Magic Number Algorithm- (188+231+264)/3=**227 µSeconds**

min_support = 4, Transaction Count = 100



Repeat Count 10 Min Support 4

```
Time Taken By Apriori:   238 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   225 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   244 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   269 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

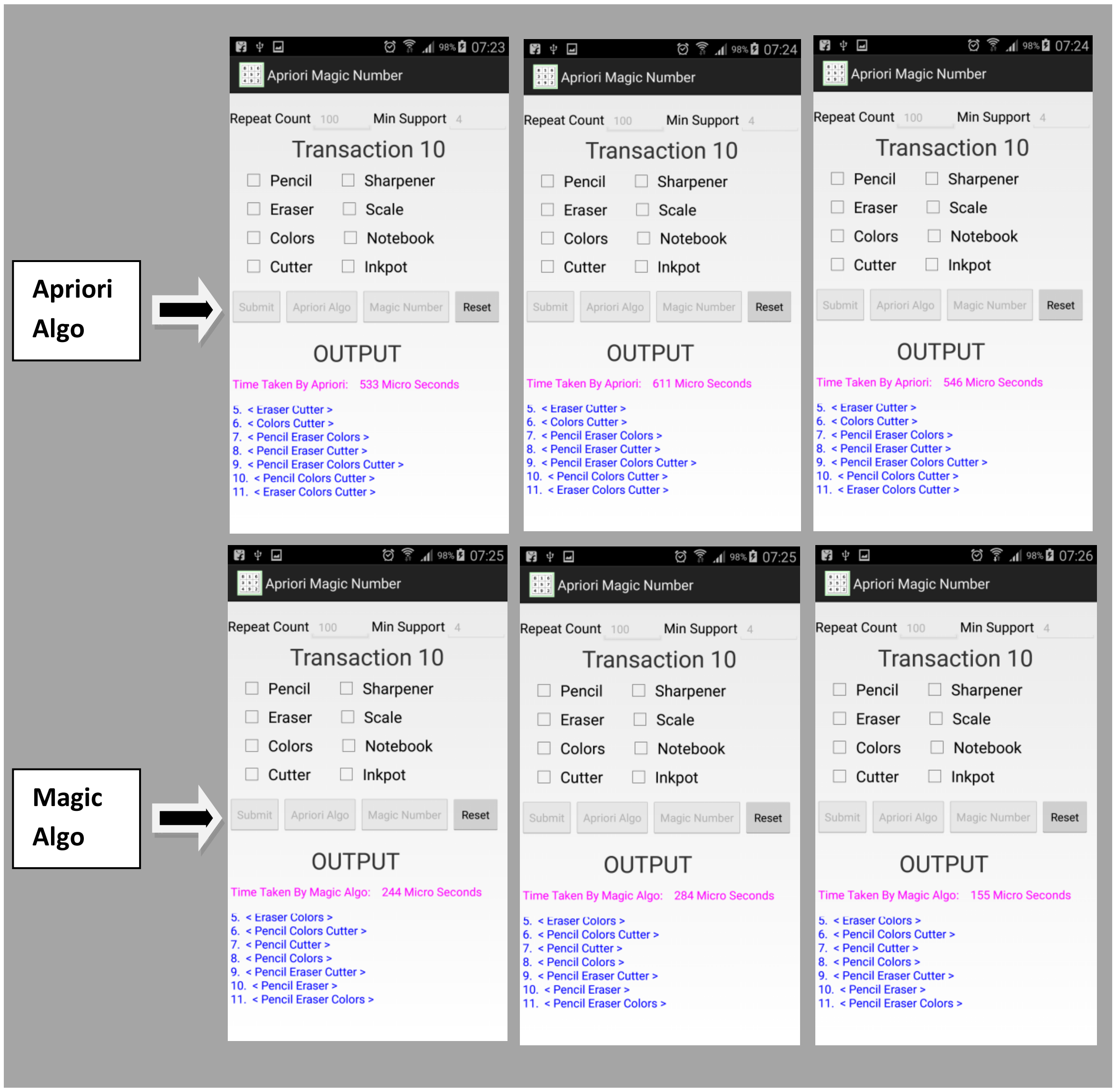
```
Time Taken By Magic Algo:   317 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

```
Time Taken By Magic Algo:   202 Micro Seconds

4.  < Eraser Cutter >
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (238+225+244)/3= **235 µSeconds**

Time taken by Magic Number Algorithm- (269+317+202)/3=**262 µSeconds**

min_support = 4, Transaction Count = 1000



```
Repeat Count 100    Min Support 4
Time Taken By Apriori:   533 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100    Min Support 4
Time Taken By Apriori:   611 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100    Min Support 4
Time Taken By Apriori:   546 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100    Min Support 4
Time Taken By Magic Algo:   244 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

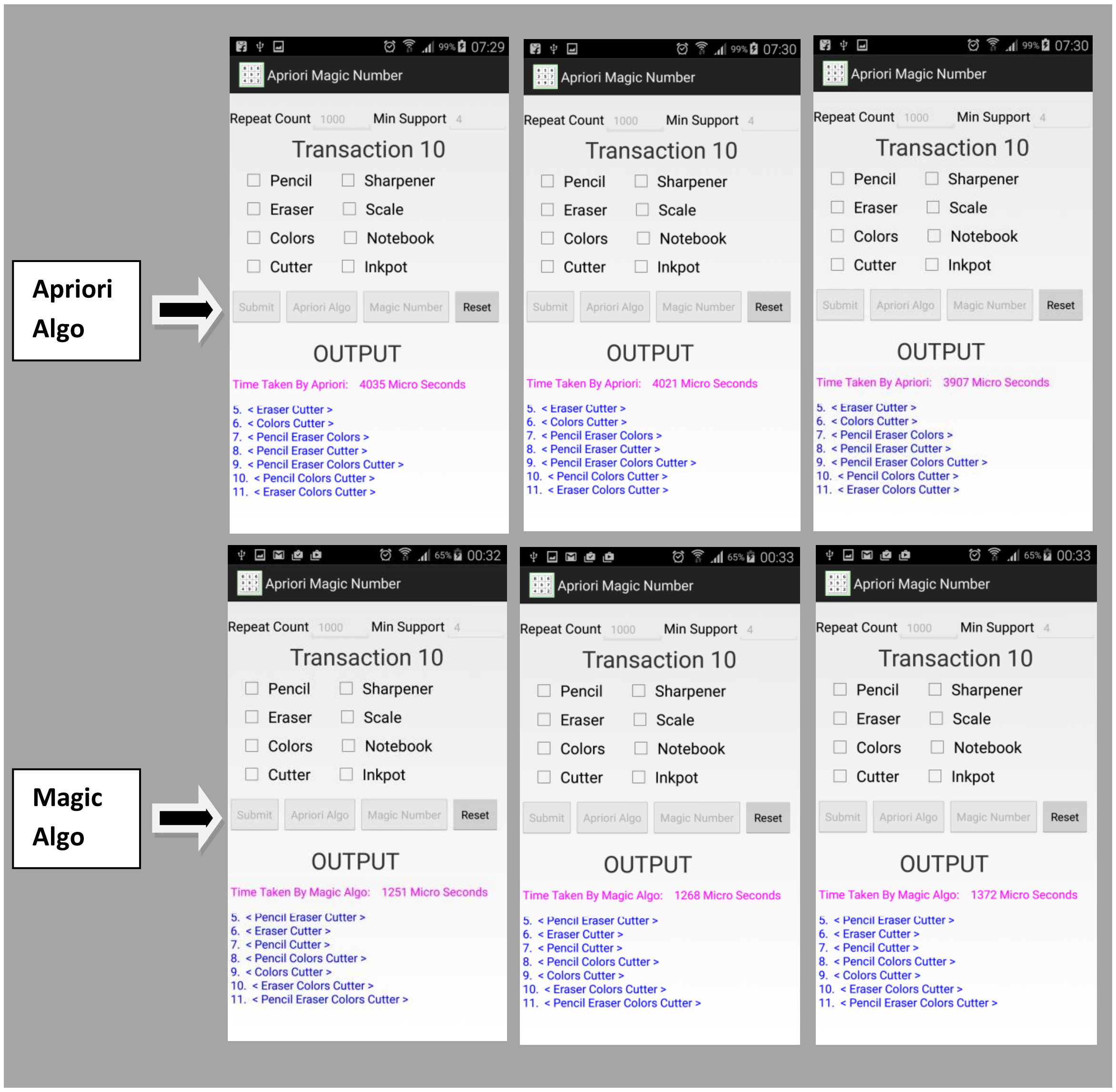
```
Repeat Count 100    Min Support 4
Time Taken By Magic Algo:   284 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

```
Repeat Count 100    Min Support 4
Time Taken By Magic Algo:   155 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (533+611+546)/3= **563 µSeconds**

Time taken by Magic Number Algorithm- (244+484+155)/3=**295 µSeconds**

min_support = 4, Transaction Count = 10,000



Repeat Count 1000 Min Support 4

```
Time Taken By Apriori:   4035 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   4021 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   3907 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1251 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

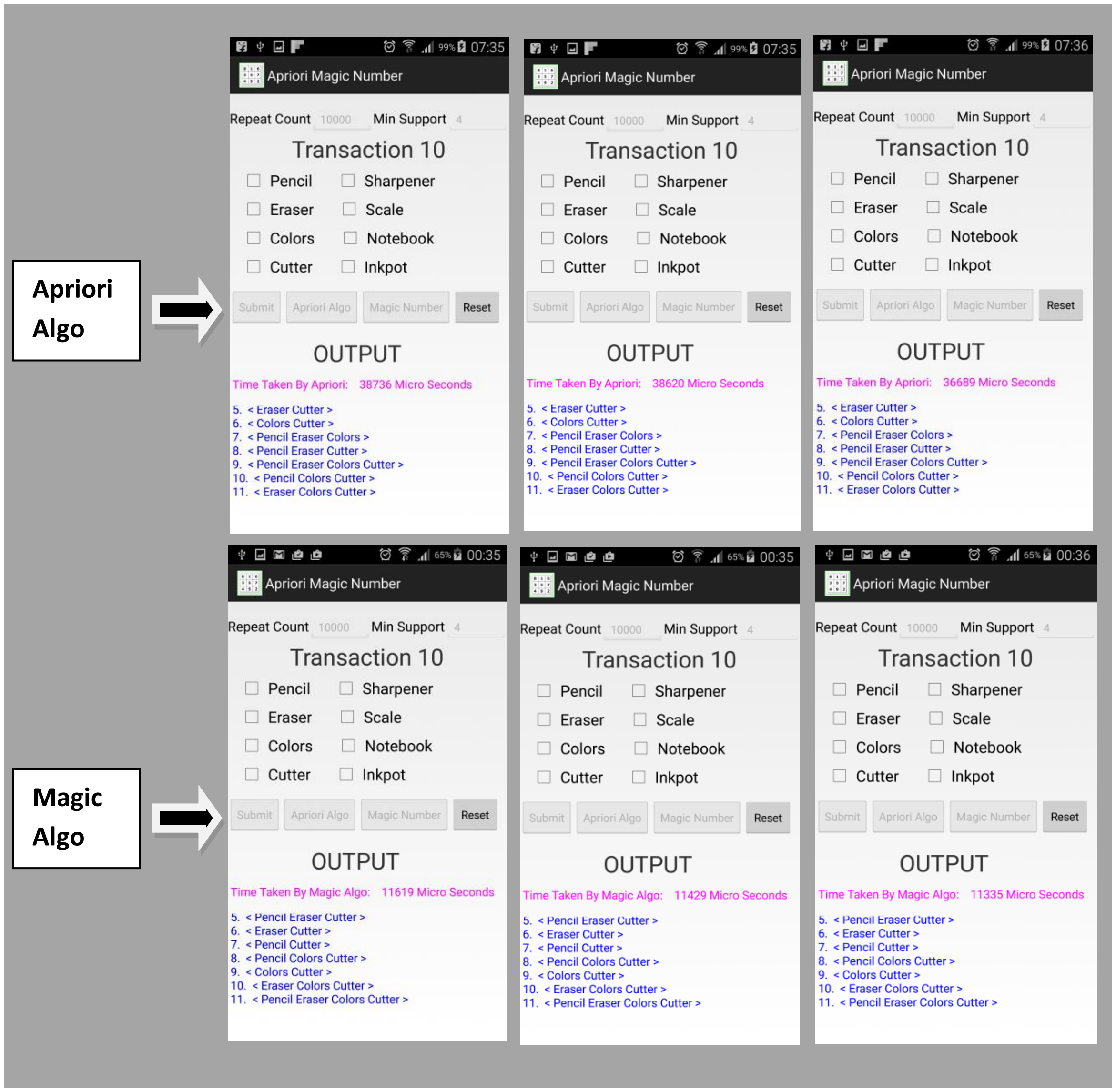
```
Time Taken By Magic Algo:   1268 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1372 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (4005+4021+3907)/3= **3977 µSeconds**

Time taken by Magic Number Algorithm- (1251+1268+1372)/3=**1297 µSeconds**

min_support = 4, Transaction Count = 100,000



```
Repeat Count 10000   Min Support 4

Time Taken By Apriori:   38736 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 4

Time Taken By Apriori:   38620 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 4

Time Taken By Apriori:   36689 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 4

Time Taken By Magic Algo:   11619 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

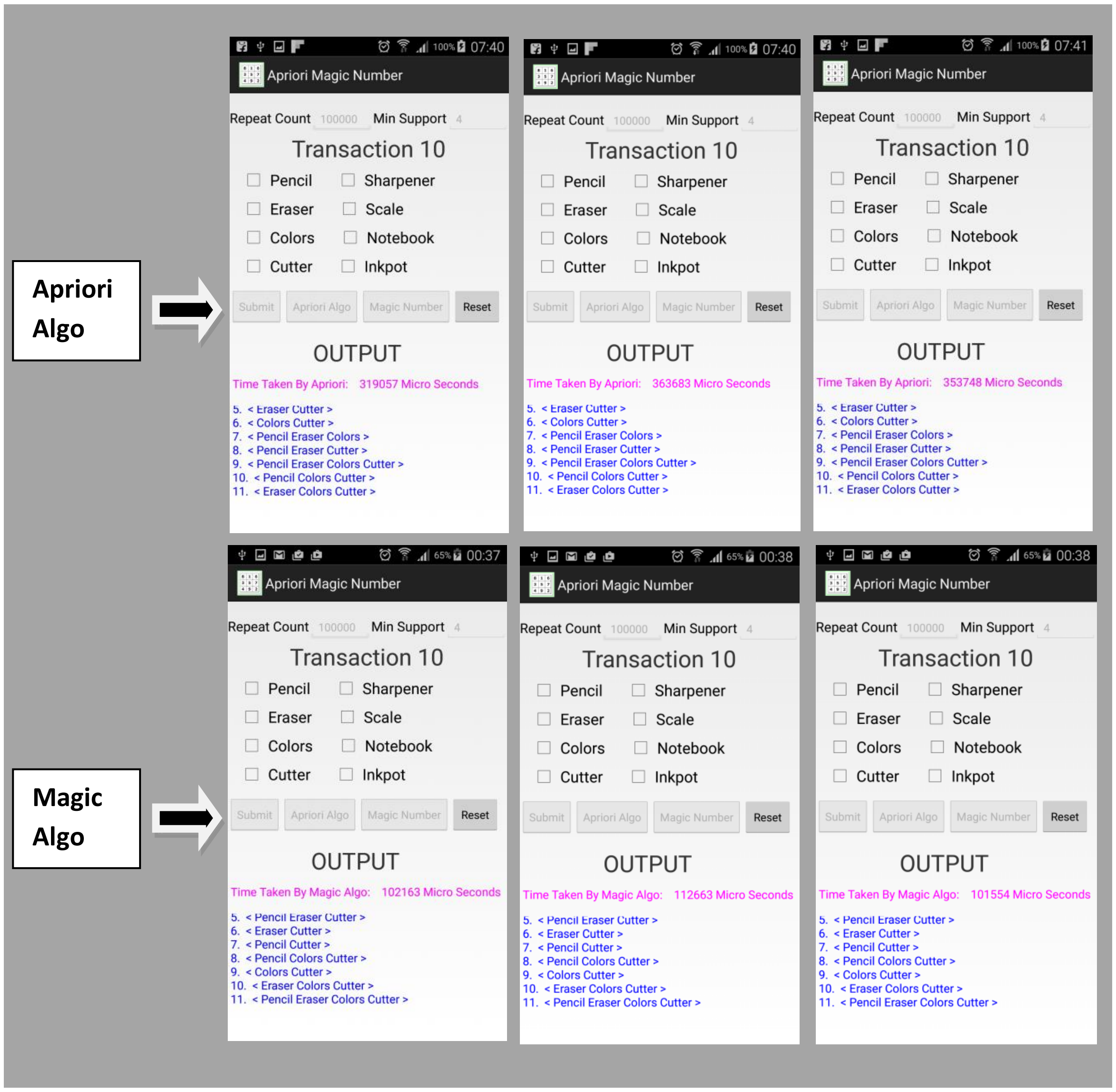
```
Repeat Count 10000   Min Support 4

Time Taken By Magic Algo:   11429 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 10000   Min Support 4

Time Taken By Magic Algo:   11335 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (38736+38620+36689)/3= **38015 µSeconds**

Time taken by Magic Number Algorithm- (11619+11429+11335)/3=**11461 µSeconds**

min_support = 4, Transaction Count = 1000,000



```
Repeat Count 100000 Min Support 4
Time Taken By Apriori:   319057 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000 Min Support 4
Time Taken By Apriori:   363683 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000 Min Support 4
Time Taken By Apriori:   353748 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000 Min Support 4
Time Taken By Magic Algo:   102163 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000 Min Support 4
Time Taken By Magic Algo:   112663 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000 Min Support 4
Time Taken By Magic Algo:   101554 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (319057+363683+353748)/3= **345496 µSeconds**

Time taken by Magic Number Algorithm- (102163+112663+101554)/3=**105460 µSeconds**

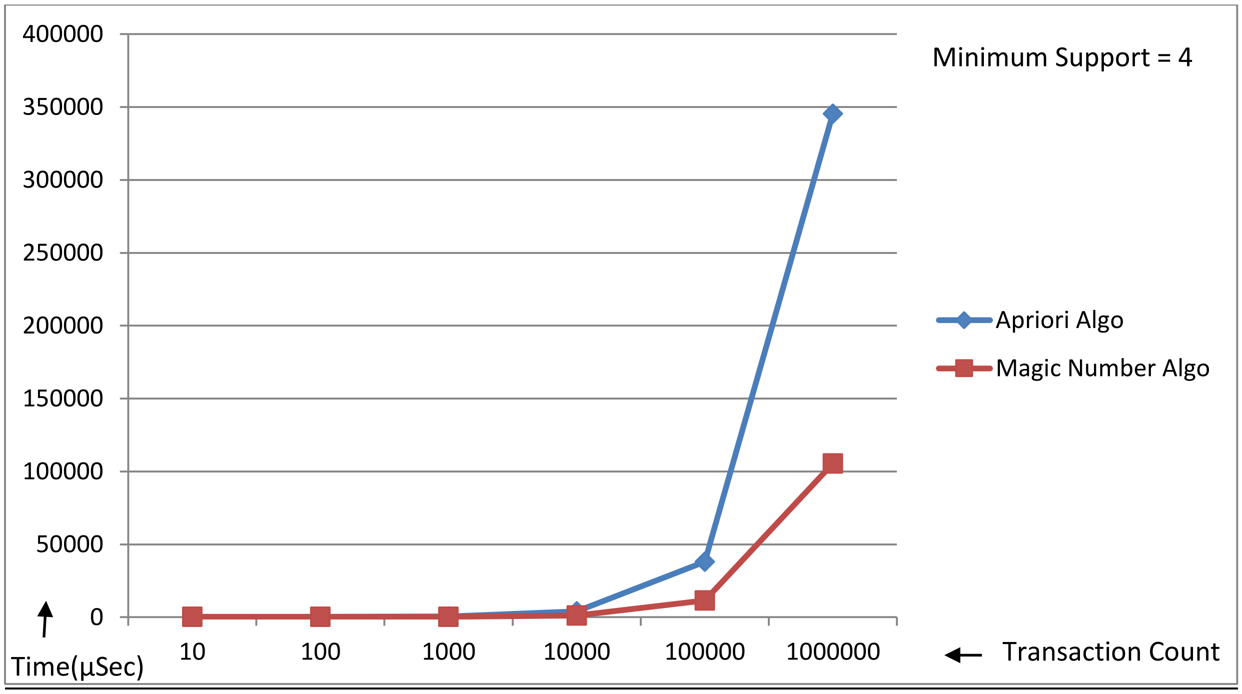
Execution Comparison In Tabular Form

Min_support = 4

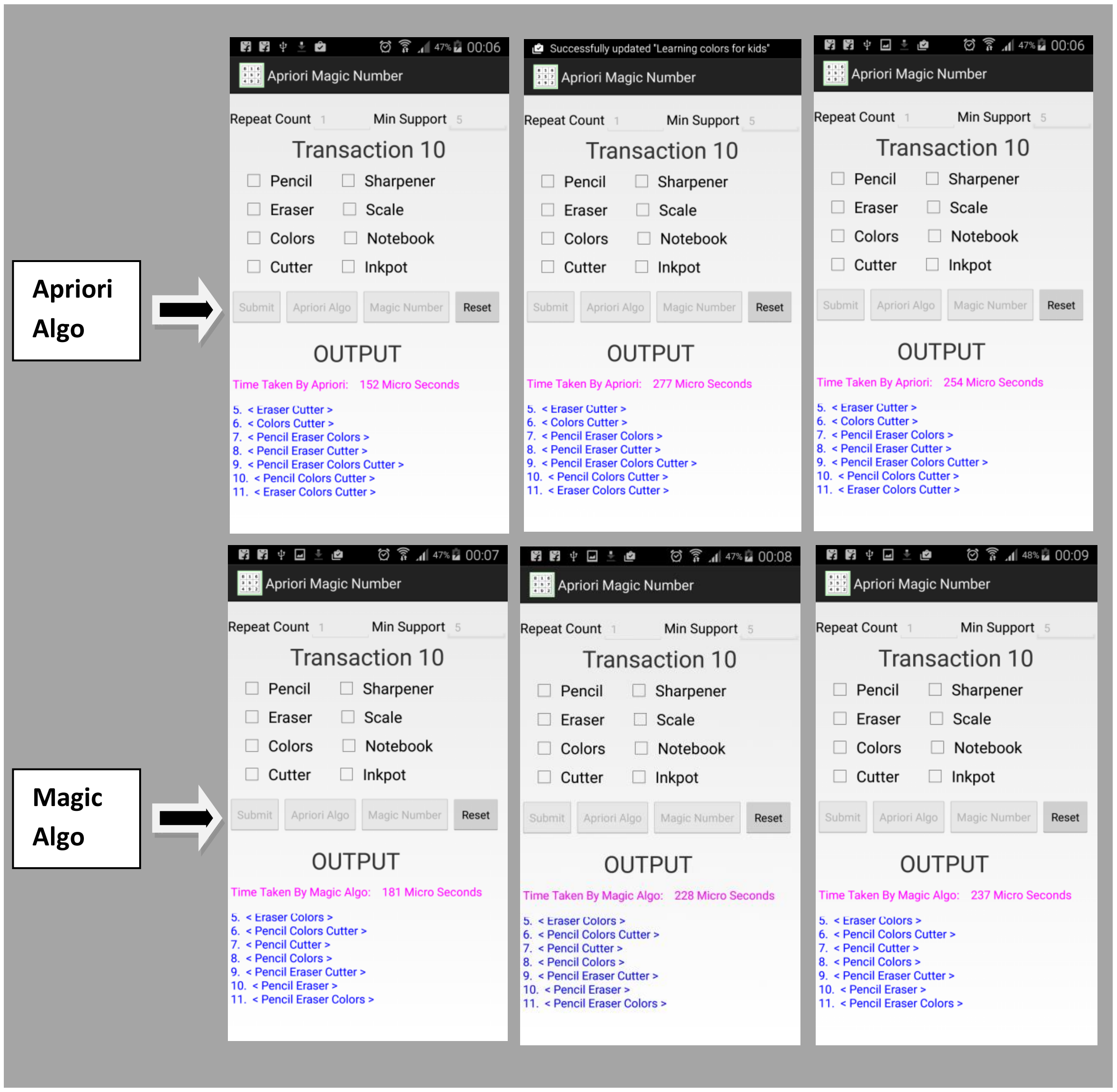
Transaction Count	Time By Apriori(µSec)	Time by Magic Number(µSec)
10	257	227
100	235	252
1000	563	295
10000	3977	1297
100000	38015	11461
1000000	345496	105460

Table 5.5 Output Table with Min Support 4

Execution Performance Comparision by Graph



min_support = 5, Transaction Count = 10



```
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Apriori:   152 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

```
Successfully updated "Learning colors for kids"
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Apriori:   277 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

```
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Apriori:   254 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

```
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Magic Algo:   181 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

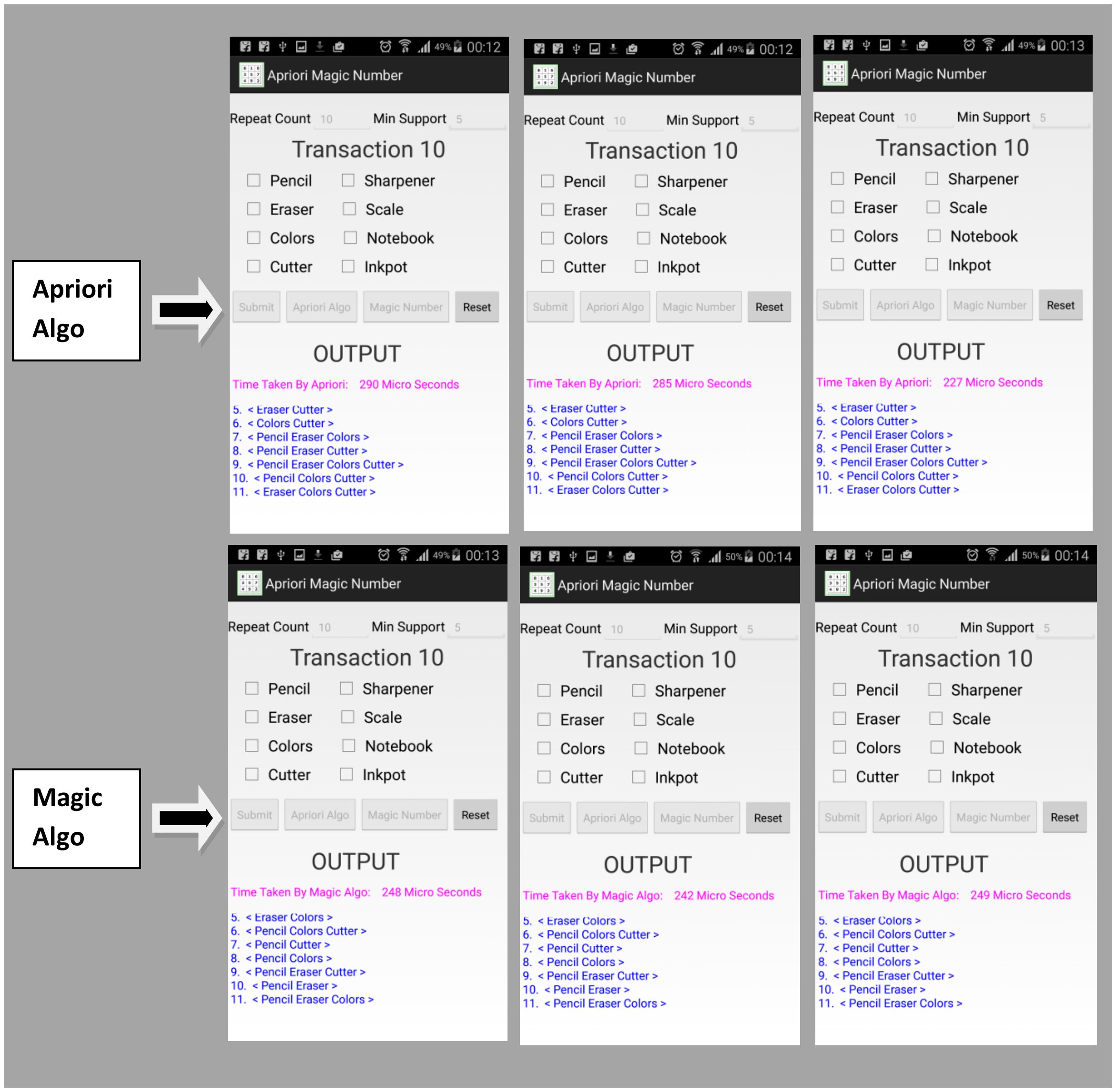
```
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Magic Algo:   228 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

```
Repeat Count 1    Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Magic Algo:   237 Micro Seconds
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (152+277+254)/3= **228 µSeconds**

Time taken by Magic Number Algorithm- (181+228+237)/3=**215 µSeconds**

min_support = 5, Transaction Count = 100



Repeat Count 10 Min Support 5

Time Taken By Apriori: 290 Micro Seconds

```
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

Repeat Count 10 Min Support 5

Time Taken By Apriori: 285 Micro Seconds

```
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

Repeat Count 10 Min Support 5

Time Taken By Apriori: 227 Micro Seconds

```
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10. < Pencil Colors Cutter >
11. < Eraser Colors Cutter >
```

Repeat Count 10 Min Support 5

Time Taken By Magic Algo: 248 Micro Seconds

```
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

Repeat Count 10 Min Support 5

Time Taken By Magic Algo: 242 Micro Seconds

```
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

Repeat Count 10 Min Support 5

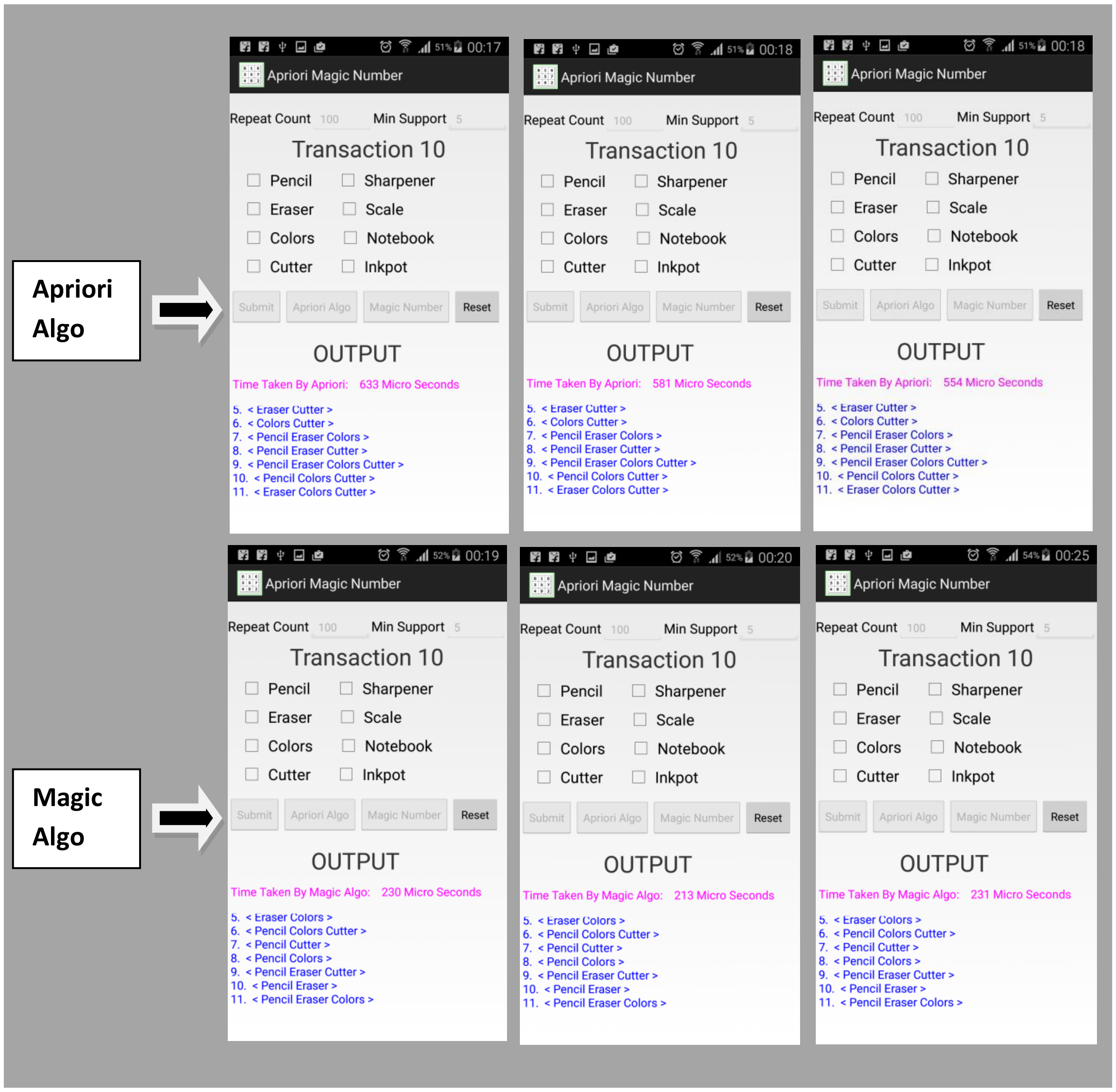
Time Taken By Magic Algo: 249 Micro Seconds

```
5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10. < Pencil Eraser >
11. < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (290+285+227)/3= **268 µSeconds**

Time taken by Magic Number Algorithm- (248+242+249)/3=**246 µSeconds**

min_support = 5, Transaction Count = 1000



Repeat Count 100 Min Support 5

```
Time Taken By Apriori:   633 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Repeat Count 100 Min Support 5

```
Time Taken By Apriori:   581 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Repeat Count 100 Min Support 5

```
Time Taken By Apriori:   554 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Repeat Count 100 Min Support 5

```
Time Taken By Magic Algo:   230 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Repeat Count 100 Min Support 5

```
Time Taken By Magic Algo:   213 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

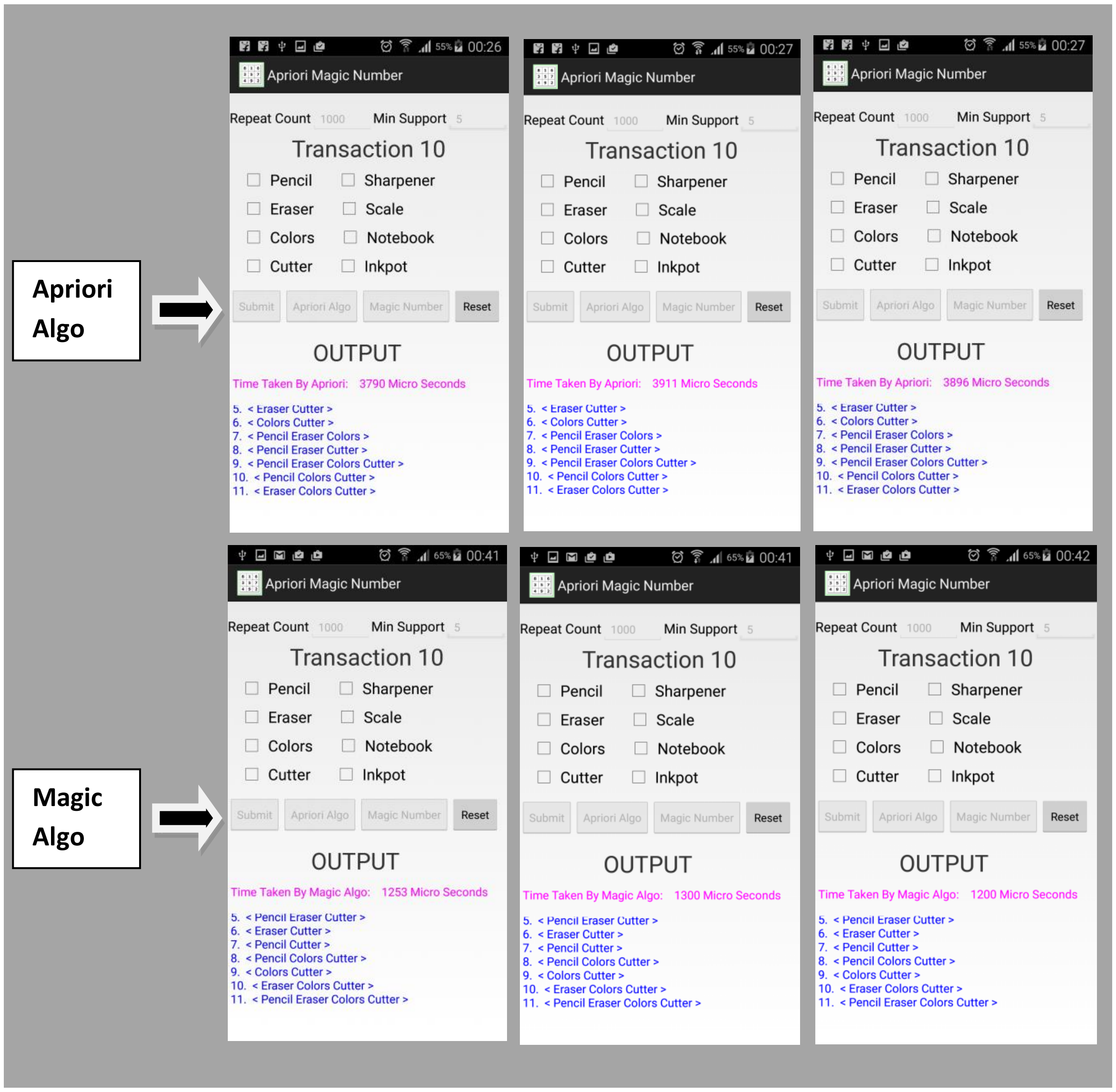
Repeat Count 100 Min Support 5

```
Time Taken By Magic Algo:   231 Micro Seconds

5.  < Eraser Colors >
6.  < Pencil Colors Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors >
9.  < Pencil Eraser Cutter >
10.  < Pencil Eraser >
11.  < Pencil Eraser Colors >
```

Time taken by Apriori Algorithm- (633+581+554)/3= **589 μSeconds**

Time taken by Magic Number Algorithm- (230+213+231)/3=**225 μSeconds**

min_support = 5, Transaction Count = 10,000



Repeat Count 1000 Min Support 5

```
Time Taken By Apriori:   3790 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   3911 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Apriori:   3896 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1253 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

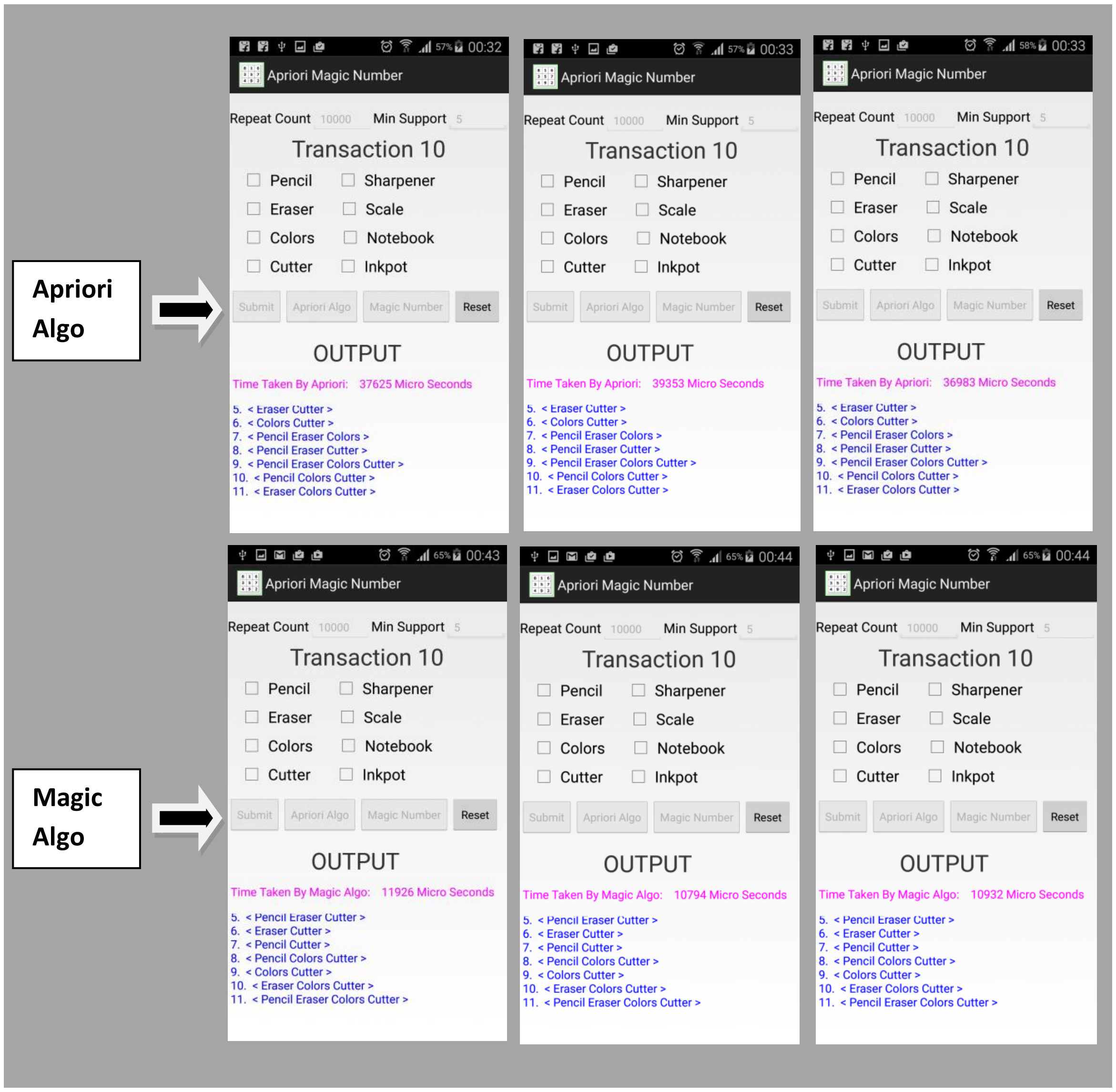
```
Time Taken By Magic Algo:   1300 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Time Taken By Magic Algo:   1200 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (3790+3911+3896)/3= **3866 µSeconds**

Time taken by Magic Number Algorithm- (1253+1300+1200)/3=**1251 µSeconds**

min_support = 5, Transaction Count = 100,000



Screenshot 1 (00:32):

```
Repeat Count 10000    Min Support 5

Time Taken By Apriori:   37625 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Screenshot 2 (00:33):

```
Repeat Count 10000    Min Support 5

Time Taken By Apriori:   39353 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Screenshot 3 (00:33):

```
Repeat Count 10000    Min Support 5

Time Taken By Apriori:   36983 Micro Seconds

5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

Screenshot 4 (00:43):

```
Repeat Count 10000    Min Support 5

Time Taken By Magic Algo:   11926 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Screenshot 5 (00:44):

```
Repeat Count 10000    Min Support 5

Time Taken By Magic Algo:   10794 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

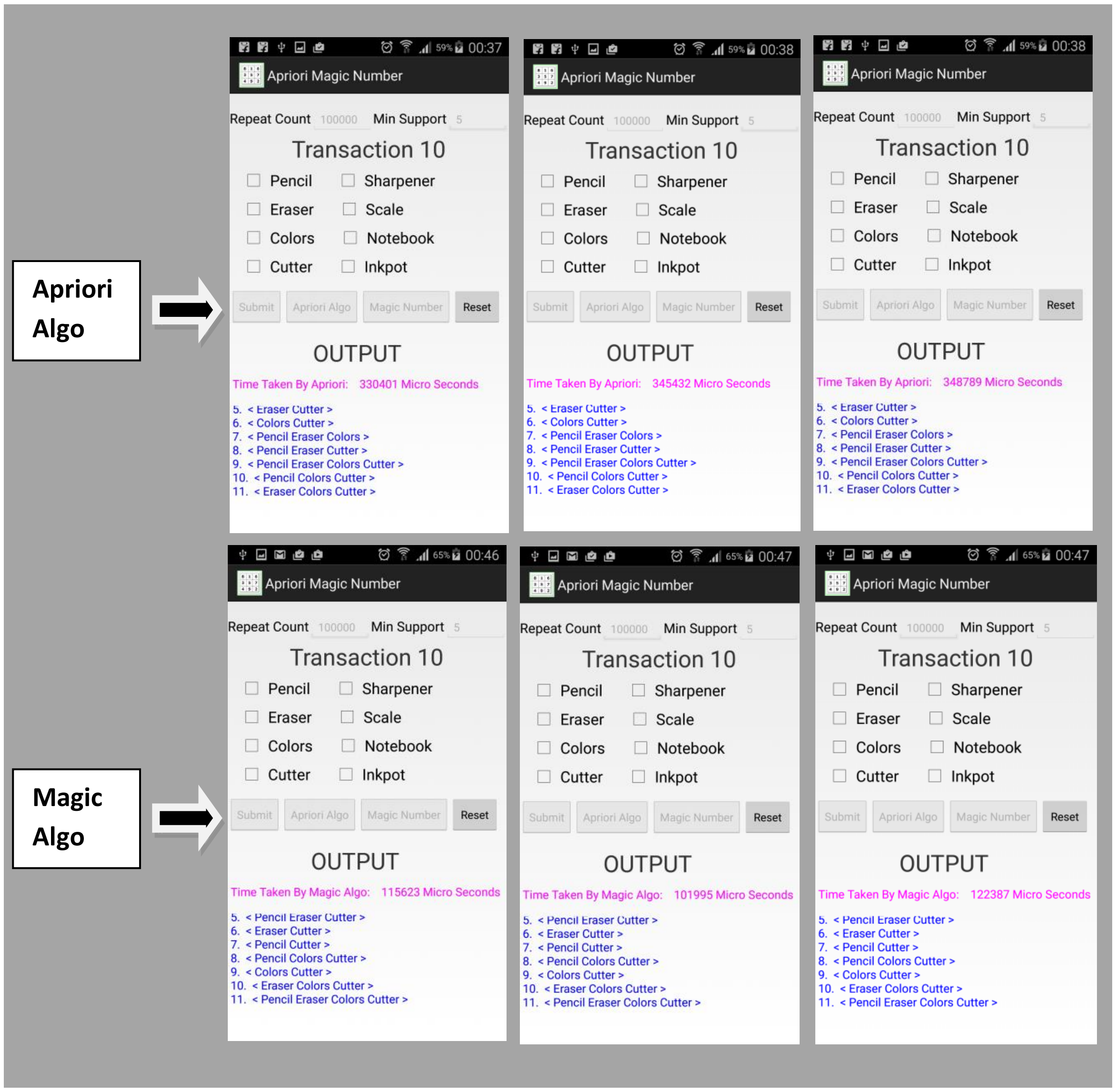
Screenshot 6 (00:44):

```
Repeat Count 10000    Min Support 5

Time Taken By Magic Algo:   10932 Micro Seconds

5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (37625+39353+36983)/3= **37987 µSeconds**

Time taken by Magic Number Algorithm- (11926+10794+10932)/3=**11217 µSeconds**

min_support = 5, Transaction Count = 1000,000



```
Repeat Count 100000   Min Support 5
Transaction 10
☐ Pencil    ☐ Sharpener
☐ Eraser    ☐ Scale
☐ Colors    ☐ Notebook
☐ Cutter    ☐ Inkpot
OUTPUT
Time Taken By Apriori:   330401 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 5
Transaction 10
OUTPUT
Time Taken By Apriori:   345432 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 5
Transaction 10
OUTPUT
Time Taken By Apriori:   348789 Micro Seconds
5.  < Eraser Cutter >
6.  < Colors Cutter >
7.  < Pencil Eraser Colors >
8.  < Pencil Eraser Cutter >
9.  < Pencil Eraser Colors Cutter >
10.  < Pencil Colors Cutter >
11.  < Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 5
Transaction 10
OUTPUT
Time Taken By Magic Algo:   115623 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 5
Transaction 10
OUTPUT
Time Taken By Magic Algo:   101995 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

```
Repeat Count 100000   Min Support 5
Transaction 10
OUTPUT
Time Taken By Magic Algo:   122387 Micro Seconds
5.  < Pencil Eraser Cutter >
6.  < Eraser Cutter >
7.  < Pencil Cutter >
8.  < Pencil Colors Cutter >
9.  < Colors Cutter >
10.  < Eraser Colors Cutter >
11.  < Pencil Eraser Colors Cutter >
```

Time taken by Apriori Algorithm- (330401+345432+348789)/3= **341540 µSeconds**

Time taken by Magic Number Algorithm- (115623+101995+122387)/3=**113335 µSeconds**

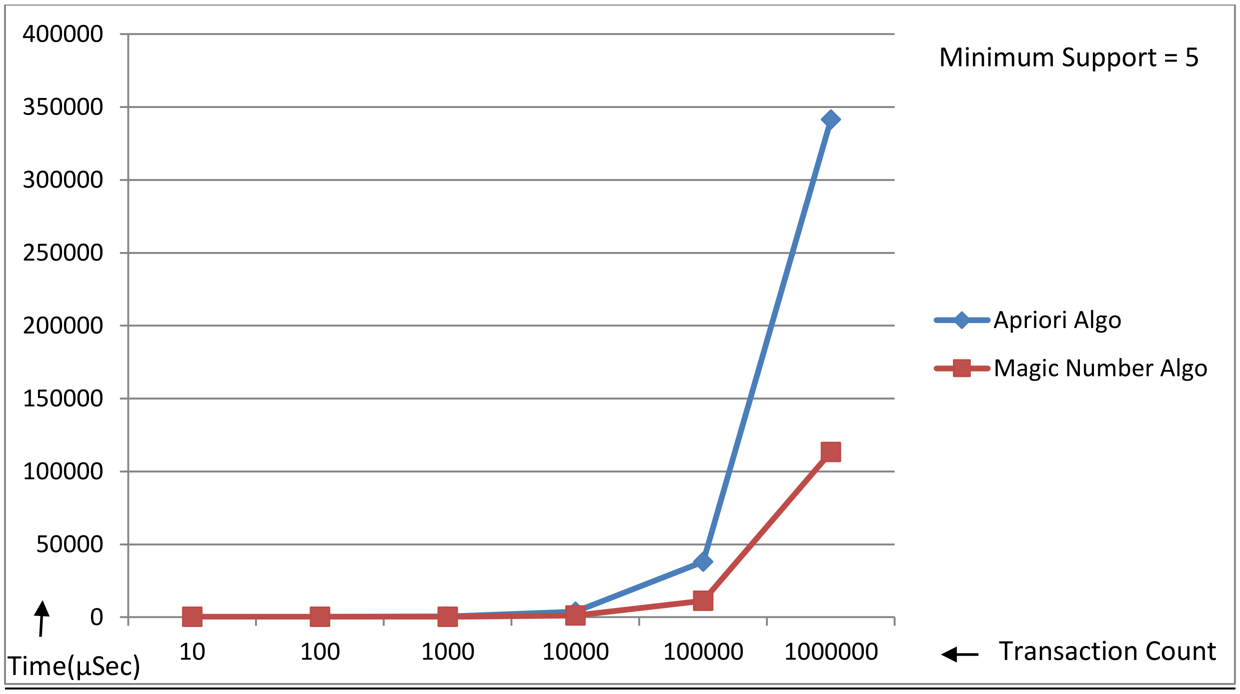
Execution Comparison In Tabular Form

Min_support = 5

Transaction Count	Time By Apriori(µSec)	Time by Magic Number(µSec)
10	228	225
100	268	246
1000	589	225
10000	3866	1251
100000	37987	11217
1000000	341540	113335

Table 5.6 Output Table with Min Support 5

Execution Performance Comparision by Graph



Chapter 6

6. CONCLUSION & FUTURE WORK

Conclusion:

Proposed Magic Number algorithm is verified by programming simulation using Android Mobile Programming and is able to extract the correct information of frequent patterns. Performance of Magic number algorithm is comparable with Apriori when there are less number of Transactions. When there is high count of transactions then Magic Number Algorithm performs better than Apriori Algorithm.

Limitation:

Our simulation for Magic Number algorithm is done with purchasable items count = 8. Magic number algorithm gives better performance as compared to Apriori because it limits the size of operational database upto range of magic sum. Magic sum range is determined by count of purchasable items. Therefore if purchasable items count is increased Magic Number algorithm may prove inferior than Apriori Algorithm in context of execution time.

Future scope :

- Determine Space complexity of this algorithm wrt Apriori
- Time & Space complexity of this algorithm wrt to other algorithm like FP Growth
- Further improvement in algorithm performance

REFERENCES

[1] R. Agrawal, T. Imielinski and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," in *ACM SIGMOD International Conference on management of data*, Washington DC, 1993.

[2] R. Agrawal, C. Faloutsos and A. Swami, "Efficient similarity search in sequence databases," in *Fourth International Conference*, Chicago, October 1993.

[3] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in *20th International Conference on Very Large Data Bases*, Santiago, Chile, September 1994.

[4] J. Han, J. Pei and Y. Yin, "Mining Frequent Patterns without Candidate Generation," in *ACM SIGMOD international conference on management of data*, Dallas, 2000.

[5] J. Han, J. Pei, Y. Yin and R. Mao, "Mining Frequent Patterns without Candidate Generation:A Frequent-Pattern Tree Approach," *Data Mining and Knowledge Discovery,* vol. 8, no. 1, pp. 53-87, 2004.

[6] H. Maurice and A. Swami, "Set-Oriented Mining for Association Rules in Relational Databases," in *Eleventh International Conference on Data Engineering*, Taipei, 1995.

[7] S. Zhang, Z. Du and J. Wang, "New Techniques for Mining Frequent Patterns in Unordered Trees," *Cybernetics, IEEE Transactions on,* vol. 45, no. 6, pp. 1113 - 1125, 2014.

[8] Y. Xu, J. X. Yu, G. Liu and H. Lu, "From Path Tree To Frequent Patterns: A Framework for Mining Frequent Patterns," in *Conference on Data Mining (ICDM 2002),* Maebashi City, Japan, 2002.

[9] T. Patel, M. Panchal, D. Ladumor, J. Kapadia, P. Desai, A. Prajapati and R. Prajapati, "An Analytical Study of Various Frequent Itemset Mining Algorithms," *Research Journal of Computer and Information Technology Sciences,* vol. 1, no. 2, pp. 6-9, 2013.

[10] A. Raorane, R. Kulkarni and B. Jitkar, "Association Rule – Extracting Knowledge Using Market Basket Analysis," *Research Journal of Recent Sciences,* vol. 1(2), no. 2, pp. 19-27, 2012.

[11] R. K. Ahir and M. B. Ahir, "ALGORITHMS FOR MINING FREQUENT: A COMPARATIVE STUDY," *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 2, no. 12, 2013.

[12] C. Borgelt, "Frequent item set mining," *Data Mining and Knowledge Discovery,* vol. 2, no. 6, pp. 437-456, 2012.

[13] T. S. Patel and K. R. Amin, "A New Approach to Mine Frequent Itemsets," *ISCA Journal of Engineering Sciences,* vol. 1, no. July, pp. 14-18, 2012.

[14] S. Pramod and O. Vyas, "Frequent Item set Mining Algorithm," *International Journal of Computer Applications,* vol. 1, no. 15, pp. 86-91, 2010.

[15] L. Wang, D. Cheung, R. Cheng, S. D. Lee and X. S. Yang, "Efficient Mining Of Frequent Item Sets On Large Uncertain Databases," *Knowledge And Data Engineering,* vol. 24, no. 12, pp. 2170-2183, 2012.

[16] R. R. Naik and J. Mankar, "Mining Frequent Itemsets from Uncertain," *International Journal of Engineering Trends & Technology in Computer Science,* vol. 2, no. 2, 2013.