SHORT QUESTIONS (5pts each)
Write- in the spaces provided - SHORT answers to the following questions

Q1 : Describe shortly Apriori Analysis; type of data, goals and types of applications.

Data: transactional data base
If a relational data base is used, it must be transformed into a set of transactions.

Goals: find set of frequent itemsets (Apriori Algorithm), find set of strong association rules (if needed).

Main Applications: improve shelving patterns in big stores, find buying patterns, improve target marketing by connecting clients data with their buying patterns.

Q2: Describe main STEPS in Apriori PROCESS

Two main steps:

1. Use of Apriori Algorithm to find the set frequent itemsets (associations) defined by user fixed support count (parameter). This is the end of the Apriori Algorithm stage.
2. Fix the confidence parameter c and find the set of strong association rules (rules with confidence greater or equal to c)
Q3: Give examples of Single-dimensional and Multi-dimensional Association Rules

**Single Dimensional:** *one predicate* used, for example in store application “buy”

\[\text{buy}(x, \text{bread}) \land \text{buy}(x, \text{milk}) \implies \text{buy}(x, \text{vodka})\]

**Inter-dimensional:** *two or more predicates* used, *no repeating predicate*. It is used when for example the associations between clients data and store transactions are needed

\[\text{age}(x, \text{old}) \land \text{income}(x, \text{small}) \implies \text{buy}(x, \text{milk})\]

**Hybrid-dimensional:** *two or more predicates* used with *repeating predicates*. It is used when for example the associations between clients data and store transactions are needed

\[\text{income}(x, \text{small}) \land \text{buy}(x, \text{bread}) \implies \text{buy}(x, \text{milk}) \land \text{age}(x, \text{old})\]

Q4: Describe principles of **Genetic Algorithms** and list types **GA** encodings and basic operators and parameters

**GA** – search and optimization

**Encodings:** binary, permutation, tree, and many others

**Basic Operators:** Initialization, Selection, Recombination, Reproduction, Termination

**Fitness Function**

**Reproduction:** crossover, mutation, elitism
Parameters: crossover probability, mutation probability

PROBLEM 1. (Classification by Association) - 15pts

1. Use the TRAIN data to find the set of classification rules by the use of Apriori Algorithm.
   Do not need to compute confidence, i.e. your rules do not need to be strong. Fix Min Support = 3
2. Test the rules with the TEST data

\[
\begin{array}{ccc}
\text{TRAIN} & \text{TEST} \\
\hline
\text{Record} & a_1 & a_2 & C \\
0_1 & 1 & 1 & 1 \\
0_2 & 0 & 0 & 0 \\
0_3 & 0 & 1 & 0 \\
0_4 & 0 & 0 & 0 \\
0_5 & 1 & 1 & 1 \\
0_6 & 1 & 1 & 0 \\
0_7 & 0 & 0 & 0 \\
0_8 & 1 & 0 & 1 \\
\end{array}
\]

Solution (almost complete) in mid2Review Lecture
Items I5, I6 represent CLASS attribute and the

Frequent items set needed to form Classification Rules is

\[ L = \{\{1,5\}, \{2,6\}, \{3,5\}, \{1,3,5\}\} \]

Rules (short hand)

\[ 1 \rightarrow 5, \quad 3 \rightarrow 5, \quad 1\&3 \rightarrow 5, \quad 2 \rightarrow 6 \]
Rules (predicate form) – for TESTING - MUST be Presented

\[ A_1(x, 0) \rightarrow C(x, 0), \quad A_2(x, 0) \rightarrow C(x, 0), \]
\[ A_1(x, 0) \& A_2(x, 0) \rightarrow C(x, 0), \]
\[ A_1(x, 1) \rightarrow C(x, 1) \]

Predictive Accuracy by verification

PROBLEM 2. (Multi-dimensional Association Rules) - 20pts

Consider a data base

\[ F = \{F_1, F_2, F_3, F_4, F_5\} \]

of food store transactions, where  \( F_1 = \{\text{Bread, Endive}\} \)
\( F_2 = \{\text{Bread, Milk}\} \),  \( F_3 = \{\text{Bread, Milk, Endive}\} \),
\( F_4 = \{\text{Bread, Endive}\} \),  \( F_5 = \{\text{Endive}\} \)

All transactions were processed with the store Cards and we have the following information (simplified) about the clients.

\( F_1 \)- lives in SBrook, Income=High
\( F_2 \) – Income=High
\( F_3 \)- lives in SBrook
\( F_4 \) - lives in SBrook, Income=High
\( F_5 \)- Income=High

1. COMBINE this information with transactions from \( F \) into one transactional data base and use the Apriori Process to find Single and Multi-dimensional Association Rules.
2. Fix minsupport = 3
3. Your rules do not need to be strong
1. Combined TRANSACTIONAL Data

<table>
<thead>
<tr>
<th></th>
<th>I1 SBrook</th>
<th>I2 IncHigh</th>
<th>I3 Bread</th>
<th>I4 Milk</th>
<th>I5 Endive</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>T2</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>T3</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>T4</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>T5</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Sc</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

Observe that this is my order of items; they can be ordered differently

We write transactions using only numbers for items

$T_1 = \{1,2,3,5\}, \ T_2 = \{2,3,4\}, \ T_3 = \{1,3,4,5\}, \ T_4 = \{1,2,3,5\}, \ T_5 = \{2,5\}$

**Frequent 1-item set** \( L_1 \) is

\[ L_1 = \{ \{1\}, \{2\}, \{3\}, \{5\} \} \]

**Join Step**: \( C_k \) (candidates) is generated by joining \( L_{k-1} \) with itself

2-item CANDIDATES SET \( C_2 \) with support count is

\[ \{1,2\}(2), \ {1,3}(3), \ {1,5}(3), \ {2,3}(3), \ {2,5}(3), \ {3,5}(3) \]

**Prune Step**: Any \((k-1)\)-itemset that is not frequent cannot be a subset of a frequent \( k \)-itemset
Frequent 2-item set  L2 is

\[ L_2 = \{\{1, 3\}, \{1, 5\}, \{2, 3\}, \{2, 5\}, \{3, 5\}\} \]

3-item CANDIDATES SET  C3 with support count is

\[ \{1, 3, 5\}(3), \{2, 3, 5\}(3) \]

Frequent 3-item set  L3 is

\[ L = \{\{1, 3, 5\}, \{2, 3, 5\}\} \]

END of Apriori

FREQUENT ITEM SET for ASSOCIATION RULES

\[ \text{FR} = \{1, 3\}, \{1, 5\}, \{2, 3\}, \{2, 5\}, \{3, 5\}, \{1, 3, 5\}, \{2, 3, 5\}\]  

SINGLE – DIMENTIONAL RULES are the rules involving only 3, 4, 5, Ie, rules formed out of \{3, 5\}.

There are two of them:

\[ \text{buys}(x, \text{bread}) \rightarrow \text{buys}(x, \text{endive}) \text{ and } \text{buys}(x, \text{endive}) \rightarrow \text{buys}(x, \text{bread}) \]

MULTI – DIMENTIONAL RULES are the rules involving 3, 4, 5, and 1, 2 Ie, rules formed out of \{1, 3, 5\} and \{2, 3, 5\}.

There are many of them.
They all are **Hybrid-dimension** rules, as they must involve 3, 5 that represent the same predicate “buys”

For example:

\[
\text{livein} (x, \text{SBrook}) \& \text{buys}(x, \text{bread}) \rightarrow \text{buys}(x, \text{endive})
\]

\[
\text{income} (x, \text{high}) \& \text{buys}(x, \text{bread}) \rightarrow \text{buys}(x, \text{endive})
\]

\[
\text{buys}(x, \text{endive}) \rightarrow \text{livein} (x, \text{SBrook}) \& \text{buys}(x, \text{endive})
\]

\[
\text{buys}(x, \text{bread}) \rightarrow \text{income} (x, \text{high}) \& \text{buys}(x, \text{endive})
\]

**PROBLEM 3. (Genetic Algorithm) – 15pts**

Here is an initial data table \( D \)

<table>
<thead>
<tr>
<th>Income</th>
<th>Student</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>No</td>
<td>Fair</td>
</tr>
<tr>
<td>high</td>
<td>yes</td>
<td>Fair</td>
</tr>
<tr>
<td>low</td>
<td>Yes</td>
<td>Fair</td>
</tr>
<tr>
<td>medium</td>
<td>Yes</td>
<td>Excellent</td>
</tr>
<tr>
<td>low</td>
<td>no</td>
<td>Fair</td>
</tr>
</tbody>
</table>

**GOAL** of the GA is to **transform** the data table \( D \) into a data table \( TD \) with the **fitness function** \( F = 0 \) for all chromosomes representing \( TD \)

**Part 1. (5pts)**

WRITE a set of Binary Encoding Chromosomes representing the Data Table \( D \). This is your **INITIAL POPULATION IP**

EVALUATE the Fitness Function **(Definition 2)** for all chromosomes in **IP**
Part 2. (10pts)
1. CREATE ONE generations P1 of your INITIAL POPULATION IP using GA operators of Selection, Single Point Crossover, Single Point Mutation for Single Point Crossover and Fitness Function defined below

Use the random selection of parents for crossover.

Definition 1
Given a chromosome Ch = c1c2c3c4 c5c6 c7 use as a cross point a point after c4, i.e. use the following single cross point

\[ c1,c2,c3,c4, | c5,c6, c7 \]

Definition 2
We define a fitness function F as follows:

For any chromosome Ch =c1c2c3c4c5c6c7 we put

\[ F(c1c2c3c4c5c6c7) = F1(c1c2c3) + F2(c4c5) + F2(c6c7) \]

Where:

\[ F1(c1,c2,c3) = 1 \text{ when only ONE 1 appears in the sequence } c1c2c3 \text{ and } F1(c1c2c3) = 0 \text{ otherwise} \]

\[ F2(c4c5) = 1 \text{ when only ONE 1 appears in the sequence } c4c5 \text{ and } F2(c4c5) = 0 \text{ otherwise} \]

\[ F2(c6c7) = 1 \text{ when only ONE 1 appears in the sequence } c6c7 \text{ and } F2(c6c7) = 0 \text{ otherwise} \]

REMEMBER that you have to keep the SIZE of the populations CONSTANT

2. Write data table D1 representing your generation P1. D1 must have a format of the initial data D
**Binary Chromosomes – INITIAL POPULATION**

<table>
<thead>
<tr>
<th></th>
<th>In=h</th>
<th>In=l</th>
<th>In=m</th>
<th>S=yes</th>
<th>S=no</th>
<th>R=fair</th>
<th>R=exc</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ch1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Ch2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Ch3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Ch4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Ch5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

All chromosomes have the same fitness, so we can choose freely two pairs for reproduction.

**REPRODUCTION**
Take, for example Ch1 and Ch2: 1000110, 1001010 and get siblings S1 and S2: 1000010, 100110
Take, for example Ch3 and Ch4: 1000110, 1001010 and get siblings S3 and S4: 0111001, 0001010

**RECOMBINATION**
Population size is 5, so we have to add one more chromosome for example one parent: 0100110

**POULATION (of 5)**
1000010, 100110, 0111001, 0001010, 0100110

**MUTATION**
Mutation probability is a parameter: I choose to mutate 2 chromosomes (random)
0000010, 100110, 0111001, 0001010, 0100100
GENERATION 1

<table>
<thead>
<tr>
<th></th>
<th>In=h</th>
<th>In=l</th>
<th>In=m</th>
<th>S=Yes</th>
<th>S=no</th>
<th>R=Fair</th>
<th>R=Exc</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ch1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Ch2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Ch3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Ch4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Ch5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

The best chromosome in GENERATION 1 is 0000010

Generation 1 Data Table

<table>
<thead>
<tr>
<th>Income</th>
<th>Student</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td>Yes</td>
<td></td>
<td>Excellent</td>
</tr>
<tr>
<td>Yes</td>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td>low</td>
<td>no</td>
<td></td>
</tr>
</tbody>
</table>

Remark that for example

chromosome Ch1 0000010 does not define values of attributes Income, Student (for record 1),

chromosome Ch2 0001110 does not define values of attribute Student – as it “says”: S=no and S=Yes what is impossible