

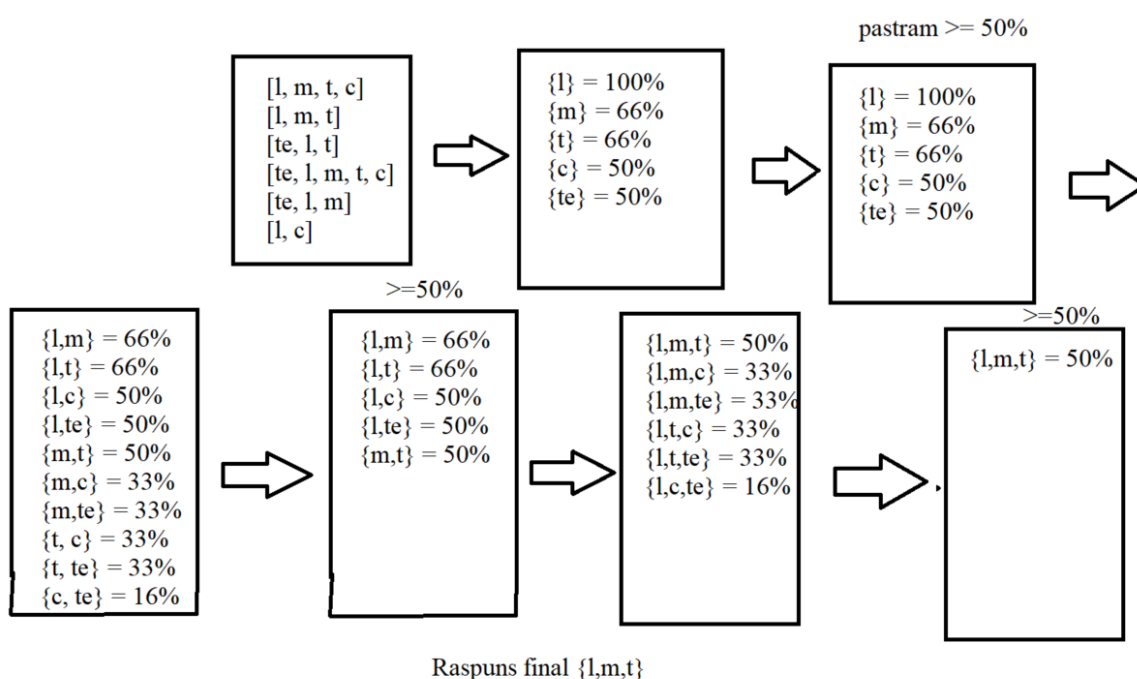
(1.5pct) Se da urmatorul set de tranzactii extrase din vanzarile unui magazin:

$D = \{(1, [\text{laptop}, \text{monitor}, \text{tableta}, \text{casti}]), (2, [\text{laptop}, \text{monitor}, \text{tableta}]), (3, [\text{telefon}, \text{laptop}, \text{tableta}]), (4, [\text{telefon}, \text{laptop}, \text{monitor}, \text{tableta}, \text{casti}]), (5, [\text{telefon}, \text{laptop}, \text{monitor}]), (6, [\text{laptop}, \text{casti}])\}$

Semnificatia este urmatoarea : D contine 6 tranzactii (de la 1 la 6), fiecare tranzactie contine o serie (continuta intre [si]) de articole cumparate (telefon,laptop, monitor, tableta, casti).

Folosind algoritmul Apriori pentru gasirea regulilor de asociere sa se gaseasca :

- (1p) Setul/seturile de elemente de lungime maxima ('large itemsets ') care sa aiba gradul de suport minim de 50% ('support').
- (0.5p) Din aceste seturi de elemente de lungime maxima ('large itemsets '), sa se genereze regulile de asociere care au gradul de incredere minima de 75% ('confidence').



$\{l, m, t\}$

$\{l, m\} \Rightarrow t$ confidence = $\text{suport}(l, m, t) / \text{suport}(l, m) = 3 / 4 = 75\%$
 $\{l, t\} \Rightarrow m$ $3 / 4 = 75\%$
 $\{m, t\} \Rightarrow l$ $3 / 3 = 100\%$
 $\{l\} \Rightarrow \{m, t\}$ $3 / 6 = 50\%$
 $\{m\} \Rightarrow \{l, t\}$ $3 / 4 = 75\%$
 $\{t\} \Rightarrow \{l, m\}$ $3 / 4 = 75\%$

[l, m, t, c]
[l, m, t]
[te, l, t]
[te, l, m, t, c]
[te, l, m]
[l, c]

Regulile de asociere cu incredere minima 75% sunt $\{l, m\} \Rightarrow t, \{l, t\} \Rightarrow m, \dots$

(3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailySales* = {*prodId*, *date*, *qty*, *price*}. La sfarsitul lunii datele trebuie consolidate intr-o tabela partitionata (o partitie pe luna) din depozitul de date cu structura *Sales* = {*year*, *month*, *prodId*, *qty*, *amount*}.

a) scrieti clauza de partitionare pentru tabela *Sales* din depozitul de date;

b) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tablei *Sales* (gestionati cazul cand *qty* poate contine valori incorecte sau lipsa, *amount* = *qty* x *price*)

c) afisati topul primelor 10 produse vandute in anul 2019 (dupa *amount*)

--a

```
CREATE TABLE my_sales(
```

```
    p_year int,
```

```
    p_month int,
```

```
    prodId int,
```

```
    qty number,
```

```
    amount float
```

```
)
```

```
PARTITION BY range (p_month) (
```

```
    PARTITION jan VALUES LESS THAN (2),
```

```
    PARTITION feb VALUES LESS THAN (3),
```

```
    PARTITION mart VALUES LESS THAN (4),
```

```
    PARTITION apr VALUES LESS THAN (5),
```

```
    PARTITION may VALUES LESS THAN (6),
```

```
    PARTITION jun VALUES LESS THAN (7),
```

```
    PARTITION jul VALUES LESS THAN (8),
```

```
    PARTITION aug VALUES LESS THAN (9),
```

```
    PARTITION sep VALUES LESS THAN (10),
```

```
    PARTITION oct VALUES LESS THAN (11),
```

```
    PARTITION nov VALUES LESS THAN (12),
```

```
    PARTITION decem VALUES LESS THAN (13)
```

```
);
```

--b

```
INSERT INTO my_sales
```

```
    (SELECT EXTRACT(YEAR FROM date), EXTRACT(MONTH FROM date), prodId,
```

```
    SUM(NVL(qty, 0)), SUM(NVL(qty, 0) * price)
```

```
    FROM DailySales
```

```
    GROUP BY EXTRACT(YEAR FROM date), EXTRACT(MONTH FROM date), prodId);
```

--c

```
SELECT prodId, p_rank FROM
```

```
    (SELECT prodId, rank() over (order by sum(amount) desc) p_rank FROM my_sales
```

```
where p_year = 2019 group by prodId)
```

```
WHERE p_rank <= 10;
```

(3pct) Modelati in MongoDB problema gestiunii *studentilor (grupa, nume, prenume)*, a *disciplinelor de curs (nume, cod, profesori)* si a *evaluarii (disciplina, student, nota)* de catre *profesori (nume, prenume, specializari)*.

a. (1.5pct) Scrieti colectiile necesare modelului, precum si structura a 1-2 documente ce pot fi inserate in colectii.

b. (1.5pct) Afisati studentii pe discipline si grupe, in ordinea descrescatoare ale notelor obtinute.

```
discipline {
```

```
  nume: string,
```

```
  cod: string,
```

```
  studenti: [
```

```
    {
```

```
      grupa: string,
```

```
      nume: string,
```

```
      prenume: string,
```

```
      evaluare: number
```

```
    }
```

```
  ]
```

```
}
```

```
profesori {
```

```
  nume: string,
```

```
  prenume: string,
```

```
  specializari: [string]
```

```
}
```

```
db.discipline.insertMany([
```

```
  {
```

```
    {
```

```
      nume: "ECBD",
```

```
      cod: "cs16",
```

```
      studenti: [
```

```
        {
```

```
          nume: "a1",
```

```
          prenume: "b1",
```

```
          grupa: "1409B",
```

```
          evaluare: 10
```

```
        },
```

```
        {
```

```
          nume: "a2",
```

```
          prenume: "b2",
```

```
          grupa: "1409B",
```

```
          evaluare: 10
```

```
}
]
},
{
    nume: "PBD",
    cod: "cs15",
    studenti: [
        {
            nume: "a3",
            prenume: "b3",
            grupa: "1409A",
            evaluare: 8
        },
        {
            nume: "a1",
            prenume: "b1",
            grupa: "1409B",
            evaluare: 9
        }
    ]
},
{
    nume: "APD",
    cod: "cs18",
    studenti: [
        {
            nume: "a1",
            prenume: "b1",
            grupa: "1409B",
            evaluare: 6
        },
        {
            nume: "a2",
            prenume: "b2",
            grupa: "1409B",
            evaluare: 10
        },
        {
            nume: "a3",
            prenume: "b3",
            grupa: "1409B",
            evaluare: 7
        }
    ]
}
```

```
    },  
  ])  
}
```

```
db.discipline.aggregate(  
  [  
    {$unwind: "$studenti"},  
    {$sort: {nume: 1, "studenti.grupa": 1, "studenti.evaluare": -1}},  
    {$project: {_id: 0, grupa: "$studenti.grupa", materie: "$nume", nota: "$studenti.evaluare",  
prenume: "$studenti.prenume", nume: "$studenti.nume"}},  
    {$group: {_id: {materie: "$materie", grupa: "$grupa"}, studenti: {$push: {nume: "$nume",  
prenume: "$prenume", "nota": "$nota"}}}},  
  ]  
)
```

Se considera un lant de magazine de electronice cu sedii in marile orase ale tarii. Sa se modeleze schema de tip documente (MongoDB) care sa stocheze vanzarile acestor magazine, tinand cont de urmatoarele categorii de campuri: magazin-oras-regiune, data-luna-an, categorie_produs-produs. Se stocheaza cantitatile vandute, precum si valoarea (cant, valoare).

a. (1.5pct) Scriti colectiile necesare modelului, precum si structura a 1-2 documente ce pot fi inserate in colectii.

b. (1.5pct) Scriti interogarile pentru a gasi vanzarile pentru o categoria de produse 'Telefoane mobile', pentru regiunea 'Moldova', in primele 3 luni ale anului curent.

```
vanzari  
{  
  data: Date,  
  luna: number,  
  an: number  
  produse: [  
    {  
      categorie_produs: string  
      nume: string  
      magazin: {  
        nume: string,  
        oras: string,  
        regiune: string  
      }  
      cantitate: number,  
      valoare: number  
    }  
  ]  
}
```

```
    ]  
}
```

```
db.vanzari.insertMany(  
    [  
        {  
            data: new Date('2020-03-12'),  
            luna: 3,  
            an: 2020,  
            produse: [  
                {  
                    "categorie_produș": "Telefoane mobile",  
                    "nume": "X",  
                    "cantitate": 3,  
                    "valoare": 1000,  
                    "magazin": {  
                        nume: "xiaomi",  
                        oras: "Iasi",  
                        regiune: "Moldova"  
                    }  
                },  
                {  
                    "categorie_produș": "Smart home",  
                    "nume": "Y",  
                    "cantitate": 1,  
                    "valoare": 1000,  
                    "magazin": {  
                        nume: "xiaomi",  
                        oras: "Iasi",  
                        regiune: "Moldova"  
                    }  
                }  
            ],  
        }  
    ],
```

```
    [  
        {  
            data: new Date('2020-04-01'),  
            luna: 4,  
            an: 2020,  
            produse: [  
                {  
                    "categorie_produș": "Telefoane mobile",  
                    "nume": "Y",  
                    "cantitate": 1,  
                }  
            ],  
        }  
    ],  
}
```

```

        "valoare": 2000,
        "magazin": {
            nume: "xiaomi",
            oras: "iasi",
            regiune: "Moldova"
        }
    }
}
]
}
]
)

```

```

db.vanzari.aggregate(
    [
        {$match: {$and: [
            {luna: {$gt: 0}},
            {luna: {$lt: 4}}
        ]}
        }},
        {$unwind: "$produse"},
        {$match: {$and:
            [
                {"produse.magazin.regiune": "Moldova"},
                {"produse.categorie_produs": "Telefoane mobile"},
            ]
        }},
        {$group: {_id: "$produse.categorie_produs", total: {
            $sum: {$multiply: ["$produse.cantitate", "$produse.valoare"]}
        }
        }
    ]
})

```

(3pct) Se considera un club de fotbal care are o serie de *echipe* (impartite pe *sex* si *categorii de varsta*), fiecare echipa avand unul sau mai multi *antrenori* si mai multi *sportivi*. Fiecare sportiv are o serie de *date de identificare* si o serie de *date sportive* (*masuratori fizice, abilitati, puncte slabe*).

Sa se modeleze schema de tip graf (OrientDB) care sa stocheze aceste date.

- (1.5pct) Scrieti clasele necesare modelului, precum si structura a 1-2 vertexi si edge-uri ce pot fi inserate in baza de date.
- (1.5pct) Scrieti interogari pentru a gasi echipa/echipele cu un numar de sportivi mai mare decat 10, pentru baietii cu varsta mai mica de 12 ani.

//a//structura

```

create class Echipa extends V
create class Antrenor extends V

```

```
create class Sportiv extends V
create class Date_sportive extends V
```

```
create property Echipa.nume string
create property Antrenor.nume string
create property Sportiv.nume string
create property Sportiv.prenume string
create property Sportiv.CNP string
create property Sportiv.varsta number
create property Sportiv.masFiz Number
create property Sportiv.abilitati string
create property Sportiv.pctSlabe string
```

```
create class Are extends E
create class Joaca extends E
create class Antreneaza extends E
```

```
//creare
```

```
create vertex Echipa set nume='Steaua'
create vertex Antrenor set nume='Mirel'
create vertex Sportiv set nume='Da', prenume='Nu', CNP='142', varsta=10, masFiz=50,
abilitati='da ta', pctSlabe='a ga b'
```

```
create edge Joaca from (select from Sportiv where nume='Da') to (select from Echipa where
nume='Steaua')
create edge Are from (select from Sportiv where nume='Da') to (select from Antrenor where
nume='Mirel')
create edge Antreneaza from (select from Antrenor where nume='Mirel') to (select from Echipa
where nume='Steaua')
```

```
//b
```

```
select nume FROM
(select nume, count(*) as nr, max(varsta) as v from
(select nume, in('Joaca').varsta as varsta FROM echipa unwind varsta)
group by nume) where v <= 12 AND nr >= 10
```

(1.5pct) Faceti o analiza comparativa intre un index de tip B-Tree si unul de tip Bitmap, intr-o baze de date relationala. Exemplificati.

(3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailyOrders* = {*store, date, orderId, amount*}. Tabela *Store* contine un camp *location*. La sfarsitul unei datele trebuie consolidate intr-o tabela partitionata (dupa 4 regiuni geografice) din depozitul de date cu structura *Sales* = {*year, month, location, store, amount*}.

- a) scrieti clauza de partitionare pentru tabela *Sales* din depozitul de date;
- b) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tabeli *Sales*
- c) afisati media vanzarilor pentru luna curenta si inca doua luni din urma pentru magazinele din zona 'Moldova'.

(3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailySales* = {*prodId, date, qty, amount*}. La sfarsitul lunii datele trebuie consolidate intr-o tabela partitionata (in partitii echidistante) din depozitul de date cu structura *Sales* = {*date, prodCateg, qty, amount*}.

- a) scrieti clauza de partitionare pentru tabela *Sales* din depozitul de date;
- b) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tabeli *Sales* (produsele se centralizeaza pe categorii, conform unei tabele de mapare)
- c) afisati ultimele 5 categorii de produse (dupa valoare) in primele 5 luni ale anului 2020.

(3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailySales* = {*prodId, date, qty, price*}. La sfarsitul lunii datele trebuie consolidate intr-o tabela partitionata (o partitie pe luna) din depozitul de date cu structura *Sales* = {*year, month, prodCateg, qty, amount*}.

- a) scrieti clauza de partitionare pentru tabela *Sales* din depozitul de date;
- b) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tabeli *Sales* ($amount = qty \times price$, produsele se centralizeaza pe categorii, conform unei tabele de mapare)
- c) afisati suma vanzarilor (dupa *qty*) pe categorii de produse si pe trimestrele anului 2019. Afisati si suma cumulata pe trimestre de la inceputul anului 2019.

(3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailySales* = {*prodId, date, qty, amount*}. La sfarsitul trimestrului datele trebuie consolidate intr-o tabela partitionata (o partitie pe trimestru) din depozitul de date cu structura *Sales* = {*year, trim, idProd, qty, amount*}.

- a) scrieti clauza de partitionare pentru tabela *Sales* din depozitul de date;
- b) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tabeli *Sales* (*prodId* difera de *idProd*, maparea facandu-se printr-o tabela de mapare pe baza unui cod comun, gestionati cazul cand apare un produs nou in *DailySales*)
- c) afisati cantitatile si valorile vandute, totale si medii, pe an si pe produs, impreuna cu subtotalurile corespunzatoare.

(3pct) Modelati in OrientDB (modelul graf) activitatea unui site in care *autorii* (*nume, prenume, nrarticole*) publica *articole* (*titlu, domeniu, cuvinte cheie, likes*) si *utilizatorii* (*username, email*) le apreciaza si sau le comenteaza.

- a. (1.5pct) Scrieti clasele necesare modelului, precum si structura a 1-2 vertexi, respectiv edge-uri ce pot fi inserati/inserate in baza de datei.
- b. (1.5pct) Afisati autorii cei mai apreciati, precum si articolele pe care le-au publicat acestia .

(1.5pct) Faceti o analiza comparativa intre framework-ul de agregare ("aggregate") si cel de map-reduce in MongoDB.

1.5pct) Se da urmatorul set de tranzactii extrase din vanzarile unui magazin:

$D = \{(1, [\text{adidas}, \text{tricot}, \text{short}, \text{geaca}]), (2, [\text{adidas}, \text{short}, \text{geaca}]), (3, [\text{adidas}, \text{tricot}, \text{short}]), (4, [\text{short}, \text{geaca}, \text{sapca}]), (5, [\text{adidas}, \text{tricot}, \text{short}, \text{sapca}])\}$

Semnificatia este urmatoarea : D contine 5 tranzactii (de la 1 la 5), fiecare tranzactie contine o serie (continuta intre [si]) de articole cumparate (adidas, tricou, short, geaca, sapca).

Folosind algoritmul Apriori pentru gasirea regulilor de asociere sa se gaseasca :

a. (1p) Setul/seturile de elemente de lungime maxima ('large itemsets ') care sa aiba gradul de suport minim de 50% ('support').

b (0.5p) Din aceste seturi de elemente de lungime maxima ('large itemsets '), sa se genereze regulile de asociere care au gradul de incredere minima de 75% ('confidence').

(1.5pct) Se da urmatorul set de tranzactii extrase din vanzarile unui magazin alimentar:

$D = \{(1, [\text{mere}, \text{pere}, \text{struguri}]), (2, [\text{pere}, \text{struguri}, \text{portocale}]), (3, [\text{portocale}, \text{zmeura}]), (4, [\text{mere}, \text{pere}, \text{portocale}]), (5, [\text{mere}, \text{pere}, \text{struguri}, \text{capsuni}]), (6, [\text{mere}, \text{pere}, \text{struguri}, \text{portocale}])\}$

Semnificatia este urmatoarea : D contine 6 tranzactii (de la 1 la 6), fiecare tranzactie contine o serie (continuta intre [si]) de fructe cumparate (mere, pere, struguri, portocale, zmeura, capsuni).

Folosind algoritmul Apriori pentru gasirea regulilor de asociere sa se gaseasca :

a. (1p) Setul/seturile de elemente de lungime maxima ('large itemsets ') care sa aiba gradul de suport minim de 50% ('support').

b (0.5p) Din aceste seturi de elemente de lungime maxima ('large itemsets '), sa se genereze regulile de asociere care au gradul de incredere minima de 75% ('confidence').

Dimensiunea maximă pentru fişierele noi: 100MB, ataşamente maxime: 2

(3pct) Se considera un club de fotbal care are o serie de echipe (impartite pe sex si categorii de varsta), fiecare echipa avand unul sau mai multi antrenori si mai multi sportivi. Fiecare sportiv are o serie de date de identificare si o serie de date sportive (masuratori fizice, abilitati, puncte slabe). Sa se modeleze schema de tip documente (MongoDB) care sa stocheze aceste date.

a. (1.5pct) Scrieti colectiile necesare modelului, precum si structura a 1-2 documente ce pot fi inserate in colectii.

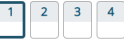
b. (1.5pct) Scrieti interogari pentru a gasi echipa/echipele cu cel mai multi sportivi, pentru baietii cu varsta mai mica de 12 ani.

(1.5pct) Faceti o analiza comparativa intre conceptele DataMining si BigData.



Extragerea cunoștințelor din baze de date (TI)

Quiz navigation



Încheiați încercarea ...

Timp rămas 0:59:45

1 Întrebare
Nu a primit
răspuns încă
Marcat din 1.50
🚩 Întrebare cu
flag

(1.5pct) Se da urmatorul set de tranzacții extrase din vanzarile unui magazin:

$D = \{ \{ \text{mere, pere, struguri} \}, \{ \text{pere, struguri, portocale} \}, \{ \text{portocale, zmeura} \}, \{ \text{mere, pere, portocale} \}, \{ \text{mere, pere, struguri, capsuni} \}, \{ \text{mere, pere, struguri, portocale} \} \}$

Folosind algoritmul Map-Reduce, contorizați fiecare tip de articol vandut din setul de tranzacții dat.

Dimensiunea maximă pentru fișierele noi: 100MB, atașamente maxime: 2



Fișiere



Puteți aduce prin drag&drop fișiere aici pentru a le adăuga.

Extragerea cunoștințelor din baze de date (TI)

(1.5pct) Faceți o analiză comparativă între un index de tip B-Tree și unul de tip Bitmap, într-o bază de date relațională. Exemplificați.

Dimensiunea maximă pentru fișierele noi: 100MB, atașamente maxime: 2



Fișiere



Puteți aduce prin drag&drop fișiere aici pentru a le adăuga.

Următoarea pagină

(3pct) Modelati in OrientDB (modelul graf) activitatea unui site in care *autorii* (*nume, prenume, nrarticole*) publica *articole* (*titlu, domeniu, cuvinte cheie, likes*) si *utilizatorii* (*username, email*) le apreciaza si sau le comenteaza.

a. (1.5pct) Scrieti clasele necesare modelului, precum si structura a 1-2 vertexi, respectiv edge-uri ce pot fi inserati/inserate in baza de datei.

b. (1.5pct) Afisati autorii cei mai apreciati, precum si articolele pe care le-au publicat acestia .

(3pct) Modelati in MongoDB activitatea unui site in care *autorii* (*nume, prenume, nrarticole*) publica *articole* (*titlu, domeniu, cuvinte cheie, likes*) si *utilizatorii* (*username, email*) le apreciaza si/sau le comenteaza.

a. (1.5pct) Scrieti colectiile necesare modelului, precum si structura a 1-2 documente ce pot fi inserate in colectii.

b. (1.5pct) Afisati autorii cei mai apreciati, precum si articolele pe care le-au publicat acestia .



Fişiere

Dimensiunea maximă pentru fişierele noi: 100MB, ataşamente maxime: 2

Puteţi aduce prin drag&drop fişiere aici pentru a le adăuga.

1. Se da următorul set de tranzacții extrase din vanzarile unui magazin alimentar: $D = ((1, [oua, paine, lapte]), (2, [cartofi, paine, bere]), (3, [oua, cartofi, paine, bere]), (4, [cartofi, bere]))$ Semnificația este următoarea : D contine 4 tranzactii (de la 1 la 4), fiecare tranzactie conține o serie (continuta intre [si]) de alimente cumparate (oua, cartofi, paine, lapte, bere).

Folosind algoritmul Apriori pentru găsirea regulilor de asociere sa se gaseasca:

a. (1p) Setul/seturile de elemente de lungime maximă ('large itemsets*') care sa aibă gradul de suport minim de 50% ('support').

b. (0.5p) Din aceste seturi de clemente de lungime maximă ('large itemsets'), sa se genereze regulile de asociere care au gradul de încredere minima de 75% ('confidence').

2. Se da următorul set de tranzacții extrase din vanzarile unui magazin: $D = ([cartofi, morcovi, rosii], [ardei, morcovi, ceapa], [cartofi, ardei, morcovi, ceapa], [ardei, ceapa])$. Folosind algoritmul Map-Reduce, contorizati fiecare tip de articole vândut din setul de tranzacții dat.

3. În baza de date relațională exista tabela de tranzacții zilnice DailyOrders (storeId, date, orderId, amount}. Tabela Store contine un camp location (zona geografica). La sfarsitul lunii, datele trebuie consolidate într-o tabela partitionata (dupa 4 regiuni geografice) și apoi subpartitionata pe luni, din depozitul de date cu structura Sales = (location, year, month, amount}.

a. (1pct) scrieți clauzele de partitionare și subpartitionare pentru tabela Sales din depozitul de date;

b. (1pct) scrieti instructiunile SQL pentru operatiile EL de actualizarea lunara a tabelii Sales (locațiile se centralizează pe regiuni, conform tabelii de mapare Store)

- c. (1pct) scrieți instrucțiunile SQL pentru operația de tip "rollup" pentru agregarea vanzarilor la nivel de trimestru.
4. Modelati pe OrientDB (modelul graf) problema gestiunii studentilor (grupa, nume, prenume), a disciplinelor de curs (nume, cod, profesori) si a evaluarii (disciplina, student, nota) de catre profesori (nume, prenume, specializari).
- (1.5pct) Scrieți clasele necesare modelului, precum si structura a 1-2 vertexi, respectiv edge-uri ce pot fi inserati/inserate in baza de date.
 - (1.5pct) Afișați student pe discipline și grupe, în ordinea descrescătoare ale notelor obținute.

4 Probleme

- 1) Analiza comparativa intre 2 concepte (1.5p)
- 2) Alg: Apriori - Map/Reduce (1.5p)
- 3) DW - ETL - Preprocesare - OLAP (cu 3 subpct) (3p)
- 4) Modelare Graph (OrientBD) / Document (MongoBD) (3p)

- Examen -

- Seria A -

1. (1.5pct) Faceti o analiza comparativa intre modelul relational si modelul NoSQL in baze de date. Luati in considerare axioma ACID, teorema CAP si modelul BASE.

2. (1.5pct) Se da urmatorul set de 6 tranzactii extrase din vanzarile zilnice ale unui magazin de fructe:

$D = \{[(mere;120), (pere;50), (struguri;75)], [(pere;30), (struguri;60), (portocale;100)], [(portocale;110), (zmeura;120)], [(mere;80), (pere;90), (portocale;50)], [(mere;65), (pere;70), (struguri;80), (capsuni;150)], [(mere;130), (pere;70), (struguri;90), (portocale;60)]\}$.

Fiecare tranzactie contine perechea formata din fructele si cantitatile corespunzatoare vandute.

Folosind algoritmul Map-Reduce, calculati cantitatea vanduta pentru fiecare categorie de fructe din setul de tranzactii dat.

3. (3pct) In baza de date relationala exista tabela de tranzactii zilnice *DailyOrders* = {*storeId*, *date*, *orderId*, *amount*}. Tabela *Store* contine un camp *location* (zona geografica). La sfarsitul lunii, datele trebuie consolidate intr-o tabela partitionata (dupa 4 regiuni geografice) si apoi subpartitionata pe luni, din depozitul de date cu structura *Sales* = {*location*, *year*, *month*, *amount*}.

a) (1pct) scrieti clauzele de partitionare si subpartitionare pentru tabela *Sales* din depozitul de date;

b) (1pct) scrieti instructiunile SQL pentru operatiile ETL de actualizarea lunara a tabeli *Sales* (locatiile se centralizeaza pe regiuni, conform tabeli de mapare *Store*)

c) (1pct) scrieti instructiunile SQL pentru operatia de tip "rollup" pentru agregarea vanzarilor la nivel de trimestru.

4. (3pct) Modelati in OrientDB (modelul graf) activitatea unui site in care autorii (nume, prenume, nrarticole) publica articole (titlu, domeniu, cuvinte cheie, likes) si utilizatorii (username, email) le apreciaza si sau le comenteaza.

a. (1.5pct) Scrieti clasele necesare modelului, precum si structura a 1-2 vertexi, respectiv edge-uri ce pot fi inserati/inserate in baza de datei.

b. (1.5pct) Afisati autorii cei mai apreciati, precum si articolele pe care le-au publicat acestia .

Curs 14 / Model Exam

4 Probleme

- 1) Analiza comparativă între 2 concepte (1,5 p)
- 2) Alg. Apriori - Map/Reduce (1,5 p) (pe nr.)
- 3) DW - ETL - Preprocesare - OLAP (cu 3 subpuncte) (3 p)
- 4) Modelare Graph (Orient DB) - Document (MongoDB) (3 p)

2. Se dă urm. set de tranzacții extrase din vânzările unui magazin alimentar:

$$D = \{(1, [\text{ouă}, \text{pâine}, \text{lapte}]), (2, [\text{cartofi}, \text{pâine}, \text{bere}]), \\ (3, [\text{ouă}, \text{cartofi}, \text{pâine}, \text{bere}]), (4, [\text{cartofi}, \text{bere}])\}$$

$D \rightarrow$ conține 4 tranzacții (de la 1 la 4), fiecare tranz. conține o secvență $[]$ de alimente cumpărate.

Cu Apriori să se găsească reguli de asocier:

- a) gășirea seturilor de elem. de lungime maximă care să aibă gradul de suport minim de 50% (support); (1 p)

- b) din aceste seturi să se genereze regulile de asocier care au gradul de încredere minimă de 75% (confidence); (0,5 p)

R:			
a)	$s - 2$	$[s, p - 2$	$s, p, c - 1$
	$p - 3$	$s, l - 1$ (nu)	$s, p, b - 1$
	$l - 1$ (nu)	$s, c - 1$ (nu)	$s, c, l - 1$
	$c - 3$	$s, b - 1$ (nu)	
	$b - 3$	$p, l - 1$ (nu)	

$$\begin{cases} p, c - 2 \\ p, b - 2 \end{cases}$$

$$\begin{aligned} & l, c - \\ & l, b - \\ & \rightarrow c, b - 3 \end{aligned}$$

Ceva nu este OK
(Trebuie să revăd că
nu e rez. OK)

Gășesc grupa maximă:

3) tabele partitionate = îți oferă posibilitatea de partitionare;

→ partitii → range

↓ după o listă (regiuni)

↓ hash (trebuie să fie distribuite între partitii);

Sales → date part. pe regiuni și după pe luni

↓
fac sumă cu rollup, slice...

(nuu o-ar face operațiunile în Oracle)

② Map Reduce

$\Delta = \{[r, m, r], [a, m, r], [r, a, m, r], [a, r]\}$

Folosind Map/Reduce → gruparea după nr. de apariție;
contorizati fiecare tip de element după nr. de apariție;

③ Tabela de tranzacții DailyOrders = {storeId, date, orderId,
amount};

Tabela Store conține → location;

La finalul lunii, datele trebuie consolidate într-o tabelă
partitionată (după 4 reg. geografice) în apoi subpart. pe luni
din dep. de date cu str. Sales = {location, year, month,
amount};

a) clauzele de partitionare și subpartitionare pt. tabela
Sales din dep. de date: (1p)

b) scrieți sint. SQL pt. op. ETL pt. actualiz. lunară a
tabelii Sales (loc. se centraliz. pe reg., conf. tab. de mapare
Store); (1p)

c) scrieți sint. SQL pt. op. de tip "rollup" pt. agregarea vânz.
la nivel de trimestru (1p);

DAILY Orders

Store ID
Order ID
Date
Amount

Store

Store ID
Location

Stores

Location
Year
Month
Amount

sau pot pune în luna '2022 05'

- * în tabela asta doar se introduce (nu se șterge);
→ part. de tip mixtă → part. principală de tip listă;
subpart. este de tip range (la nivel de lună);
↓ adăugă

~~PARTITION BY LIST (LOCATION)~~

~~SUBPARTITION BY MONTH~~

~~(PARTITION SALES_MOLDOVA ('IS', 'BT', 'SU', ...),~~

~~PARTITION SALES_MUNTENIA ('BZ', 'VR', ...),~~

a) PARTITION BY LIST (LOCATION)

SUBPARTITION BY MONTH

(PARTITION L1 VALUES ('IS', ...))

(SUBPARTITION L1-M1 VALUES LESS THAN '2022 01'),

(SUBPARTITION L1-M2 VALUES LESS THAN '2022 02'),

⋮
(SUBPARTITION L1-M12 VALUES LESS THAN '2022 12')

),
PARTITION L2 VALUES ('BZ', ...)

(

)

d) SELECT S.LOCATION, TO_CHAR(DATE, 'YYYY') AS YEAR,
 CONCAT(TO_CHAR(DATE, 'YYYY'), TO_CHAR(DATE, 'MM')) AS Month, SUM(AMOUNT)
 FROM table ----
 GROUP BY location, concat in ^{year} ~~date~~;
 (toate 3 altfel eroare)

e) pt. agregare →

SALES derive

LOC
 YEAR
 TRIM
 AMOUNT

* sau cum cu case-uri cu where pt. trimestru;
 * sau oă creez o tabelă: ~~location~~

TIMES

TIME-ID
 YEAR
 MONTH
 TRIM
 SEM

→ identific după year și month
 Pp. că am tabela Times deja populată;

SELECT S.LOCATION, S.YEAR, T.TRIM, SUM(S.AMOUNT)
 FROM SALES S, TIMES T
 WHERE S.YEAR = T.YEAR AND
 S.MONTH = T.MONTH

GROUP BY LOC, YEAR, TRIM;
 HAVING TRIM = 'T2';

4) Model graf (Orientat DB)

Gestiunea studenților (grupă, nume, prenume)

a dir. de curs (nume, cod, profesori)

a eval. (disciplină, student, notă)

de către profesori (nume, prenume, specializări);

a) Scrieți clasele necesare modelului, precum în str. a 1-2
vertexi, precum în edge-uri a pot fi inserați în BD, (1,5p)

Clase:

Studenți

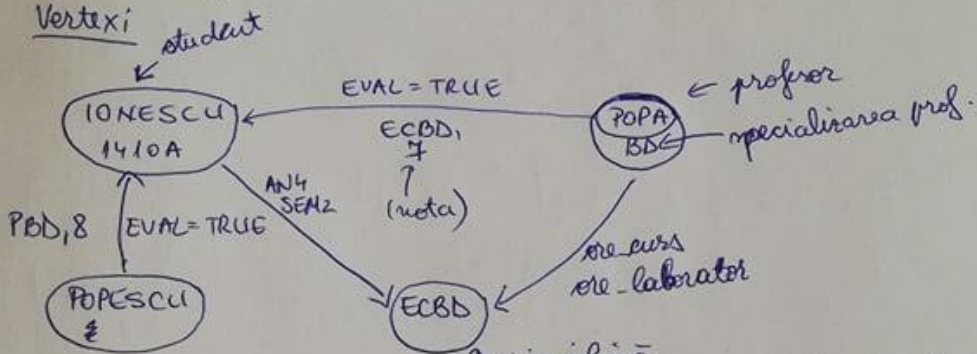
Discipline

Profesori

Studenți și Profesori menținere clasa persoană;

→ se definesc cu atribute, numele clasei;

Vertexi



Fiecare vertex trebuie să aibă edge-uri de intrare și ieșire;

Ionescu: am muniția de la Popa, Popescu și totii profesori care
il eval.; (intrări) ← un array

un array cu munițiile ce ies din vertex (discipl.
pe care le-a urmat);

* la fel pt. profesori și discipline;

* e suf. atât, doar de precizat exact atributele pe
fiecare vertex și muniție;

b) Afişați studenții pe discipline în gr., în ord. desc. ale notelor obținute; (1,5p)

x parcurse vertexii de tip student în de ~~mereu~~ pe edgeurile de tip evaluare unde se ta sf. de la stud., numele, discip. în nota (e suficient să facem atât);

ps:

```
SELECT FROM Student WHERE EVAL=TRUE  
ORDER BY NOTA NOTA DESC;  
x un punctaj mai mic (la asta → e posibil);
```