

Ex1:

Cererea SQL:

```
SELECT data_inceput, data_sfarsit, pret
```

```
FROM sejur S LEFT JOIN locatie L USING(cod_locatie)
```

```
WHERE zona == 'munte' AND s.data_inceput > TO_DATE('01-JAN-2019', 'DD-MONTH-YYYY')
```

```
UNION
```

```
(
```

```
SELECT data_inceput, data_sfarsit, pret
```

```
FROM excursie E LEFT JOIN angajat A USING(nr_buletin) LEFT JOIN OFERTA O USING(cod_oferta)
```

```
WHERE data_angajare > TO_DATE('01-MAY-2017', 'DD-MONTH-YYYY') AND job_cod = 'sofer';
```

```
INTERSECT
```

```
SELECT data_inceput, data_sfarsit, pret
```

```
FROM oferta O LEFT JOIN excursie E USING(cod_oferta)
```

```
WHERE E.nr_buletin IN(
```

```
    SELECT nr_buletin
```

```
    FROM limba
```

```
    WHERE limba = 'engleza'
```

```
INTERSECT
```

```
    SELECT nr_buletin
```

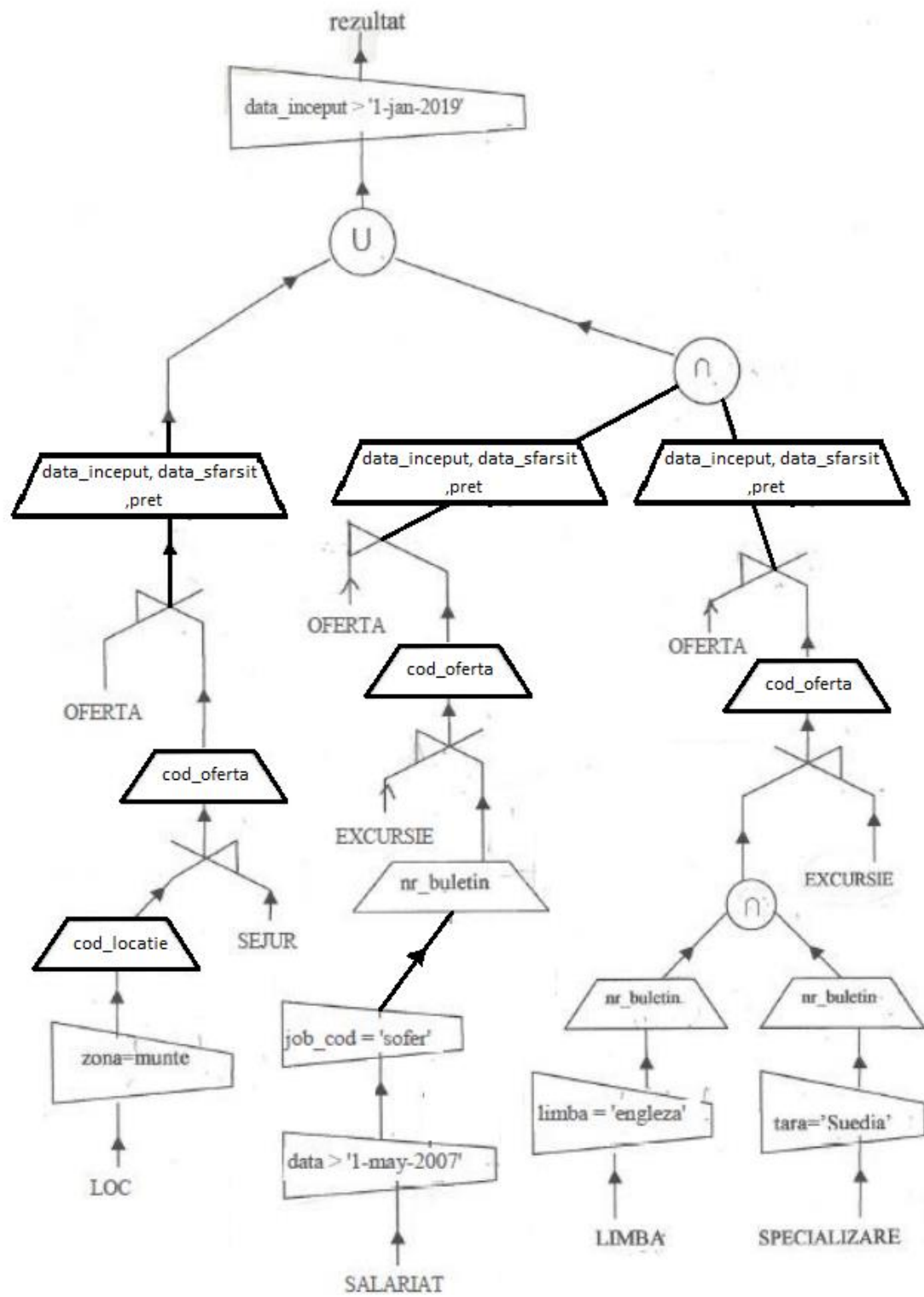
```
    FROM specializare
```

```
    WHERE tara = 'Suedia';
```

```
)
```

```
);
```

Arborele algebric ameliorat:



Optimizări aduse: după fiecare operație JOIN am pus o operație PROJECT pentru a reduce numărul de coloane folosite deci a optimiza relația (după regula de optimizare 4 din curs!). De asemenea, am șters ultimul PROJECT (se afla înainte de ultimul SELECT) deoarece nu mai era necesar!

Expresia algebrică:

R1 = SELECT(LOC, zona = 'munte')

R2 = PROJECT(R1, cod_locatie)

R3 = JOIN(R2, SEJUR, R2.cod_locatie = SEJUR.cod_locatie)

R4 = PROJECT(R3, cod_oferta)

R5 = JOIN(R4, OFERTA, R4.cod_oferta = OFERTA.cod_oferta)

REZ1 = PROJECT(R5, data_inceput, data_sfarsit, pret)

R1 = SELECT(ANGAJAT, data > '1-MAY-2007')

R2 = SELECT(R1, job_cod = 'sofer')

R3 = PROJECT(R2, nr_buletin)

R4 = JOIN(R3, EXCURSIE, R3.nr_buletin = EXCURSIE.nr_buletin)

R5 = PROJECT(R4, cod_oferta)

R6 = JOIN(R5, OFERTA, R5.nr_buletin = OFERTA.nr_buletin)

REZ2 = PROJECT(R6, data_inceput, data_sfarsit, pret)

R1 = SELECT(LIMBA, limba = 'engleza')

R2 = PROJECT(R1, nr_buletin)

R3 = SELECT(SPECIALIZARE, tara = 'Suedia')

R4 = PROJECT(R3, nr_buletin)

R5 = INTERSECT(R2, R4)

R6 = JOIN(R5, EXCURSIE, R5.nr_buletin = EXCURSIE.nr_buletin)

R7 = PROJECT(R6, cod_oferta)

R8 = JOIN(R7, OFERTA, R7.cod_oferta = OFERTA.cod_oferta)

REZ3 = PROJECT(R7, data_inceput, data_sfarsit, pret)

REZ4 = INTERSECT(REZ2, REZ3)

REZ = INTERSECT(REZ1, REZ4)

FINAL_REZ = SELECT(REZ, data_inceput : '01-JAN-2019')

Ex2:

Cererea SQL:

```
SELECT cod_contractant, nume
FROM SUBANTREPRENOR
WHERE cod_contractant IN (
    SELECT cod_contractant
    FROM LUCRARE
    WHERE UPPER(tip) = 'SPECIALIZAT' AND cod_ob_inv IN (
        SELECT cod_ob_inv
        FROM OBIECTIV_INVESTITIE
        WHERE UPPER(denumire) = 'CABANA'

        UNION

        SELECT cod_ob_inv
        FROM OBIECTIV_INVESTITIE
        WHERE UPPER(denumire) = 'CASA_VACANTA'
    )
);
```

SELECT cod_contractant, nume

FROM SUBANTREPRENOR

WHERE cod_contractant IN (

SELECT cod_contractant

FROM LUCRARE

WHERE UPPER(tip) = 'SPECIALIZAT' AND cod_ob_inv IN (

SELECT cod_ob_inv

FROM OBIECTIV_INVESTITIE

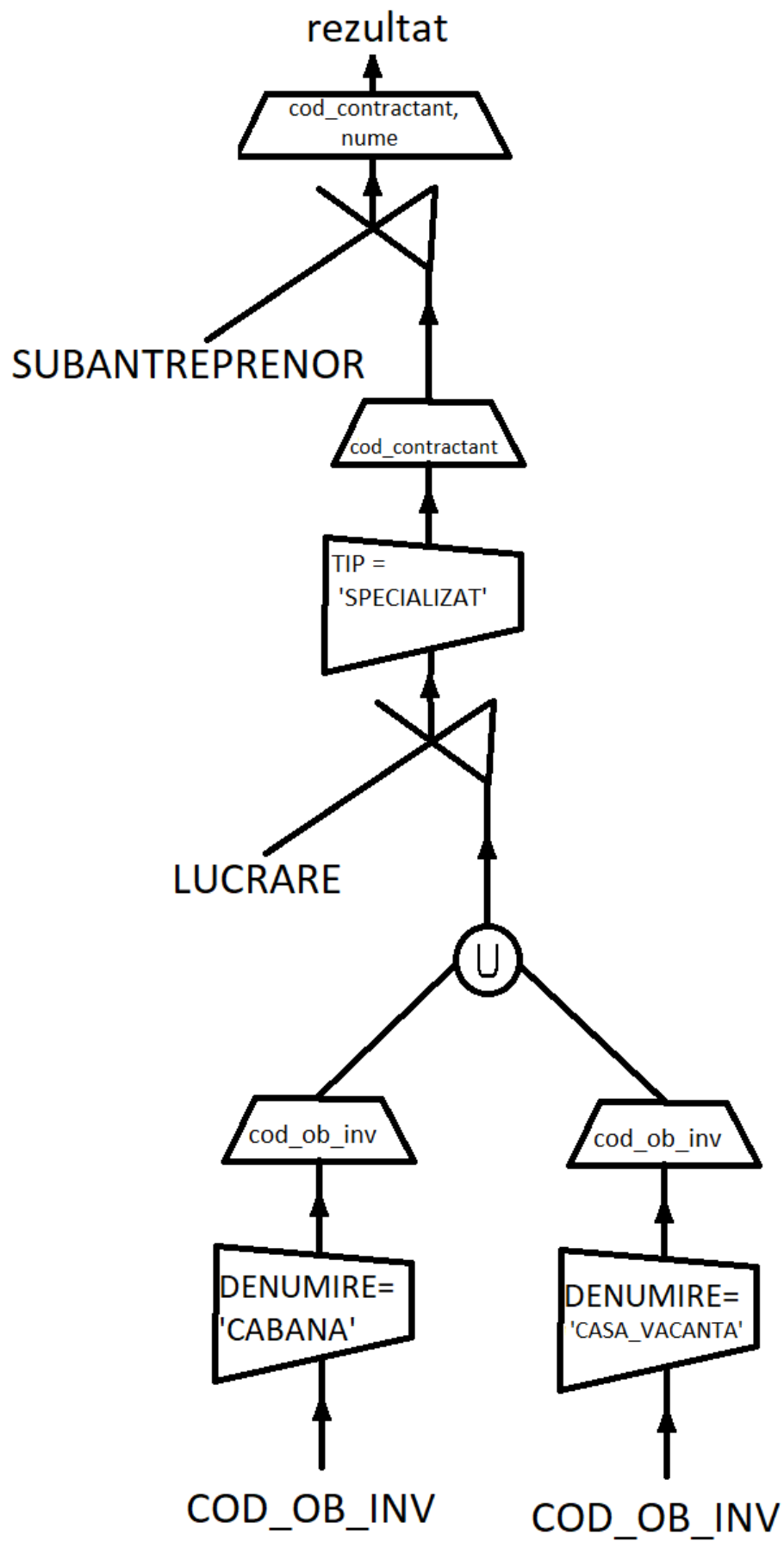
WHERE UPPER(denumire) = 'CABANA'

UNION

```
SELECT cod_ob_inv  
FROM OBJECTIV_INVESTITIE  
WHERE UPPER(denumire) = 'CASA_VACANTA'  
)
```

```
);
```

Arborele relational:



Expresie algebrică:

R1 = SELECT(COD_OB_INV, denumire = 'CABANA')

R2 = PROJECT(R1, cod_ob_inv)

R3 = SELECT(COD_OB_INV, denumire = 'CASA_VACANTA')

R4 = PROJECT(R3, cod_ob_inv)

R5 = INTERSECT(R2, R4)

R6 = JOIN(R5, LUCRARE, R5.cod_ob_inv = LUCRARE.cod_ob_inv)

R7 = SELECT(R6, tip = 'SPECIALIZAT')

R8 = PROJECT(R7, cod_contractant)

R9 = JOIN(R8, SUBANTREPRENOR, R8.cod_contractant = SUBANTREPRENOR.cod_contractant)

REZ = PROJECT(R9, cod_contractant, nume)