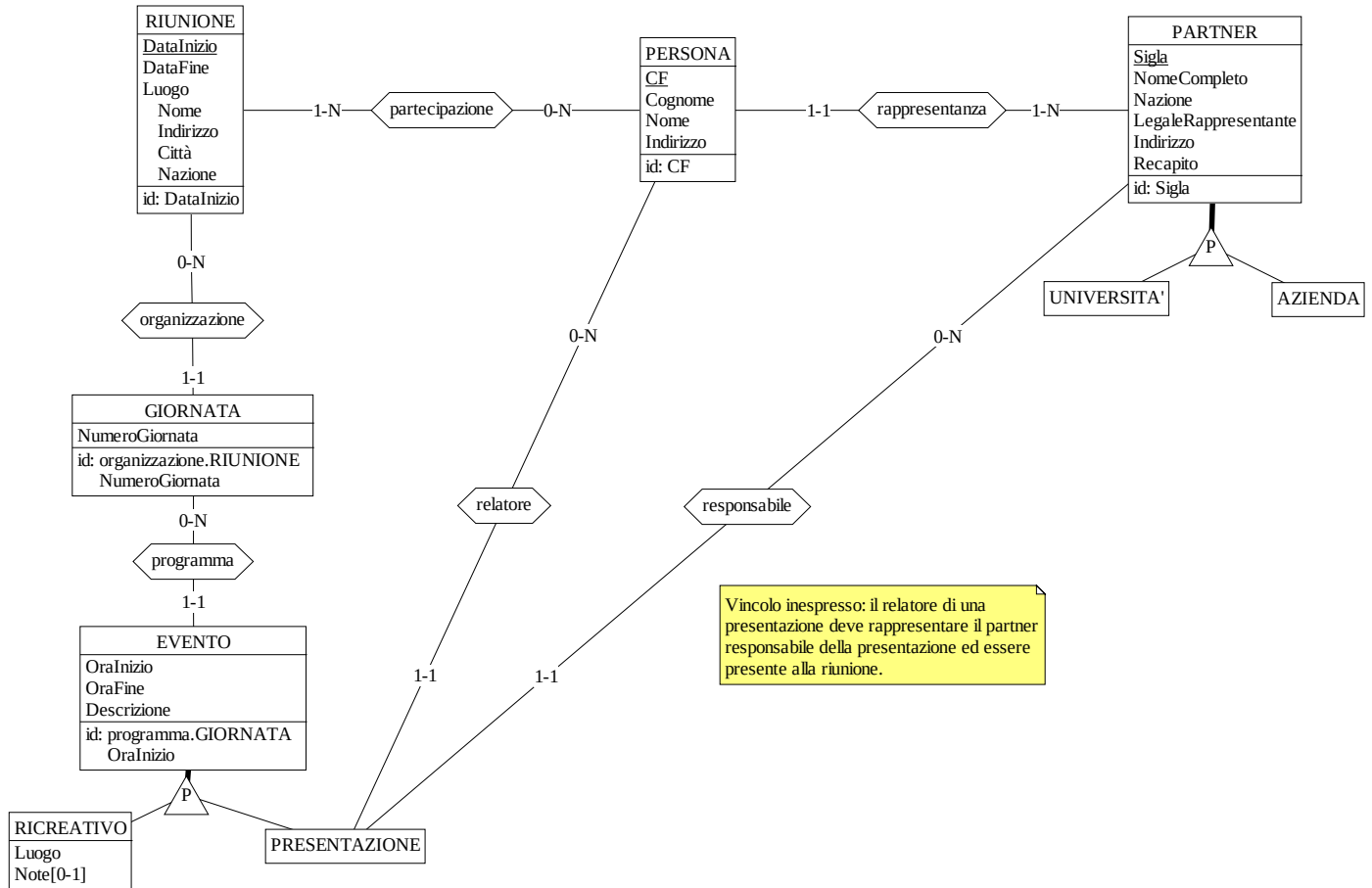


Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_ Matricola: \_\_\_\_\_

**Esercizio 1**

## Esercizio 2

Chiave: CodFotografia, CodMostra

CodFotografia → Titolo, CodFotografo, DataFotografia (Parziale)

CodFotografo → NominativoFotografo, DataNascitaFotografo, NazionalitàFotografo (Transitiva)

CodMostra → NomeMostra, CittàMostra, IndirizzoMostra, NazioneMostra, DataInizio, DataFine (Parziale)

FOTOGRAFIE (CodFotografia, Titolo, CodFotografo: FOTOGRAFI, DataFotografia)

FOTOGRAFI (CodFotografo, NominativoFotografo, DataNascitaFotografo, NazionalitàFotografo)

MOSTRE (CodMostra, NomeMostra, CittàMostra, IndirizzoMostra, NazioneMostra, DataInizio, DataFine)

ESPOSIZIONI (CodFotografia: FOTOGRAFIE, CodMostra: MOSTRE)

**Esercizio 3****Selettività dei predicati** $p1: fattura\_emessa = 1$  $p2: punto\_vendita = 'P28'$ 

$$f(p1) = \frac{1}{2} = 0.5; f(p2) = \frac{1}{50} = 0.02$$

**Record attesi**

$ER$  (expected records);  $EP$  (numero di pagine necessarie a contenere i dati attesi);  $ER_{Result}$  (numero atteso di record nel risultato finale);

$$ER = \lceil NR \times f(p1) \times f(p2) \rceil = \lceil 450000 \times 0.5 \times 0.02 \rceil = 4500$$

$$EP = \lceil \frac{4500 \times \text{len}(\text{select list})}{4000} \rceil = \lceil \frac{4500 \times 49}{4000} \rceil = 56$$

$$ER_{Result} = ER = 4500$$

**Costi**

$C_{Sort}$  (costo di ordinamento con Z-way merge sort);  $C_{Seq}$  (costo di accesso con scansione sequenziale);  $C_{punto\_vendita}$  (costo di accesso tramite indice su punto\_vendita);

$$C_{Sort} = 2 \times EP \times \lceil \log_z EP \rceil = 2 \times 56 \times \lceil \log_3 56 \rceil = 448$$

$$C_{Seq} = NP + C_{Sort} = 7677 + 448 = 8124$$

Accesso tramite l'indice costruito su *punto\_vendita*:

$$C_{punto\_vendita} = \lceil f(p2) \times NL_{punto\_vendita} \rceil + EK_{punto\_vendita} \times \Phi\left(\frac{NR}{NK_{punto\_vendita}}, NP\right) + C_{Sort} = 14 + 1 \times 5301 + 448 = 5763$$

Il piano di accesso più conveniente è quindi quello basato su indice unclustered.

**Esercizio 4**

1.  $\pi_{codCondomino, nome, cognome} \left( \left( PRESENZE \div \pi_{codRiunione} \left( \sigma_{YEAR(data)=2019} (RIUNIONI) \right) \right) \triangleright \triangleleft CONDOMINI \right)$
2. 

```
SELECT nome, cognome
FROM CONDOMINI
WHERE codCondomino NOT IN (SELECT codCondomino
                           FROM Presenze P, Riunioni R
                           P.codRiunione = R.codRiunione
                           AND data = (SELECT MAX(R1.data)
                                       FROM Riunioni R1))
```
3. 

```
SELECT nome, cognome, COUNT(*) AS numeroRiunioni
FROM Riunioni R, Condomini C
WHERE R.codRiunione = C.codRiunione
AND YEAR(data) = 2018
GROUP BY C.codCondomino, nome, cognome
HAVING COUNT(*) >=2
```
4. 

```
SELECT R.codRiunione, data, COUNT(*) AS numeroPunti, COUNT(DISTINCT codRelatore) AS numeroRelatori
FROM OrdineDelGiorno O, Riunioni R
WHERE O.codRiunione = R.codRiunione
GROUP BY R.codRiunione, data
```