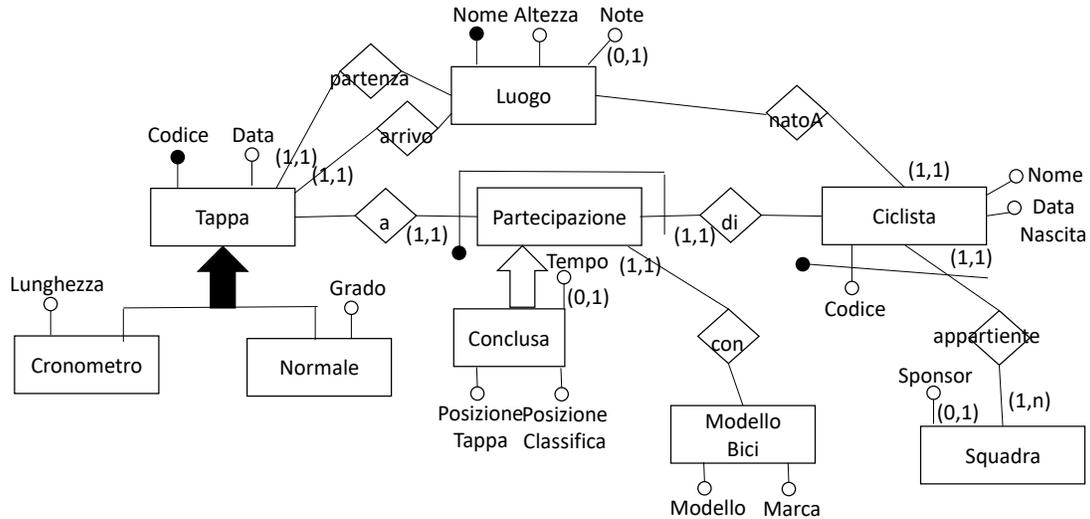


Appello di Basi di Dati – 15/09/2020
 Proff. Tiziana Catarci e Francesco Leotta - A.A. 2019-2020

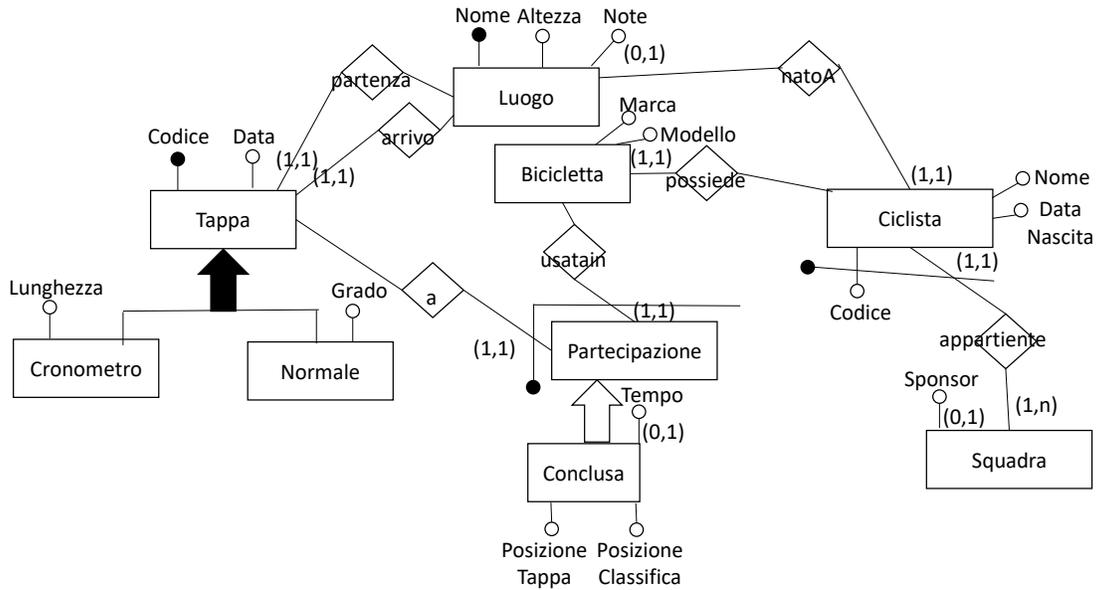
DOMANDA 1 (10 punti)



L'attributo Tempo dell'entità Conclusa è popolato solo se la tappa è di tipo a Cronometro.

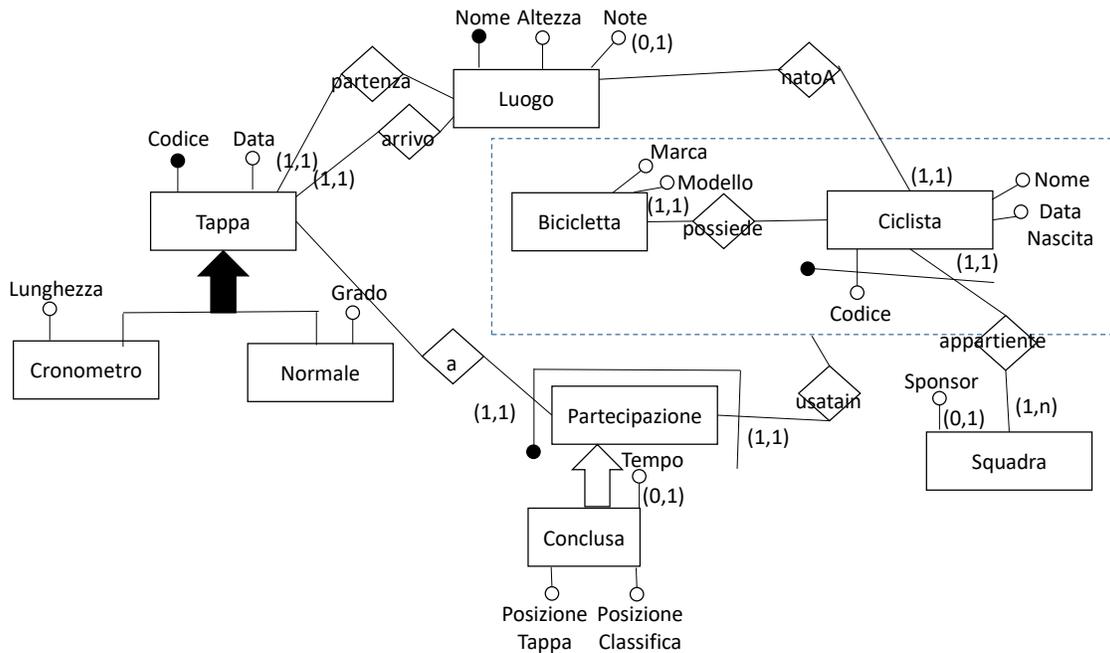
Era possibile anche modellare la partecipazione come una relazione con conclusa relazione con vincolo di ISA. In questo caso la progettazione logica si complicava per via dell'attributo opzionale (con vincolo di evitare valori nulli) in fase di ristrutturazione. Si potevano inoltre utilizzare due relazioni con Cronometro e Normale invece che una relazione con Tappa per evitare il vincolo esterno sull'attributo Tempo.

Come soluzione alternativa si poteva interpretare la bicicletta come la specifica bicicletta e non come modello di bicicletta. In questo caso la soluzione, ugualmente valida, poteva essere la seguente:



L'attributo Tempo dell'entità Conclusa è popolato solo se la tappa è di tipo a Cronometro.

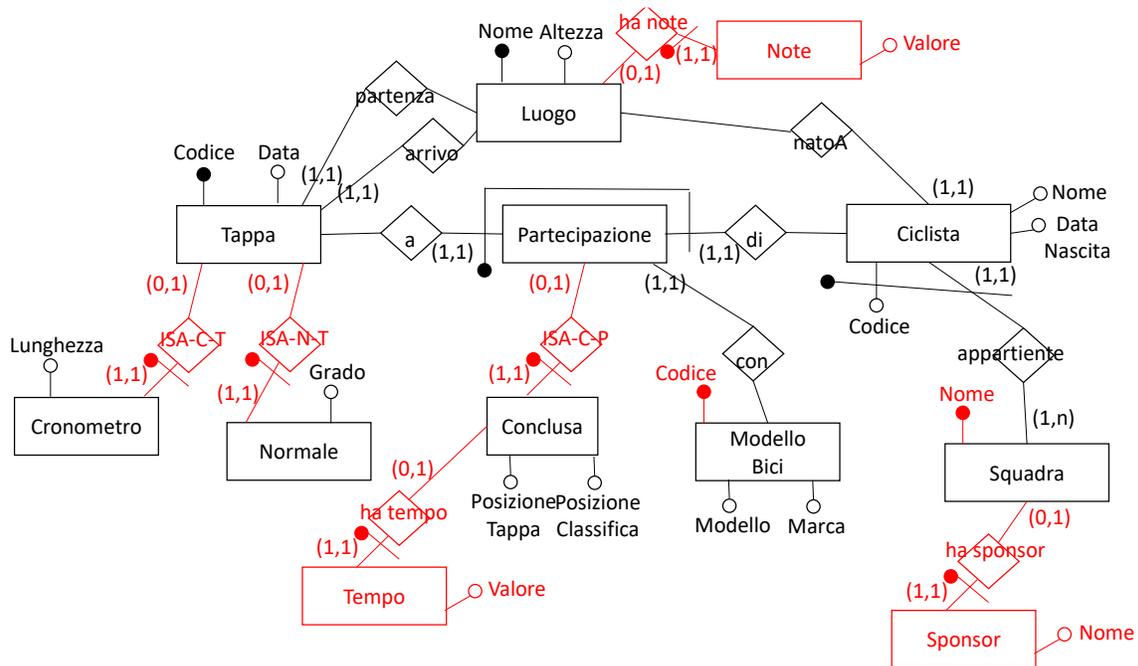
Si poteva, sempre in alternativa, utilizzare una aggregazione come segue:



L'attributo Tempo dell'entità Conclusa è popolato solo se la tappa è di tipo a Cronometro.

DOMANDA 2

La soluzione verrà proposta con riferimento alla prima soluzione indicata alla DOMANDA 1



L'attributo Tempo dell'entità Conclusa è popolato solo se la tappa è di tipo a Cronometro.

Ogni istanza di tappa è necessariamente o una istanza di cronometro o una istanza di normale.

Luogo (Nome, Altezza)

Tappa(Codice, Data, LuogoPartenza, LuogoArrivo)

FK: Tappa[LuogoPartenza] C Luogo[Nome]

FK: Tappa[LuogoArrivo] C Luogo[Nome]

Cronometro(Tappa, Lunghezza)

FK: Cronometro[Tappa] C Tappa[Codice]

Normale(Tappa, Grado)

FK: Normale[Tappa] C Tappa[Codice]

Squadra(Nome)

INC: Squadra[Nome] C Ciclista[Squadra]

Ciclista(Codice, Squadra, Nome, DataNascita, LuogoNascita)

FK: Ciclista[Squadra] C Squadra[Nome]

FK: Ciclista[LuogoNascita] CLuogo[Nome]

Note(Luogo, Valore)

FK: Note[Luogo] CLuogo[Nome]

Sponsor(Squadra, Nome)

FK: Sponsor[Squadra] CSquadra[Nome]

Partecipazione(Tappa, CodiceCiclista, SquadraCiclista, ModelloBici)

FK: Partecipazione[Tappa] CTappa[Codice]

FK: Partecipazione[CodiceCiclista, SquadraCiclista] CCiclista[Codice, Squadra]

FK: Partecipazione[ModelloBici] CModelloBici[Codice]

Conclusa(Tappa, CodiceCiclista, SquadraCiclista, PosizioneTappa, PosizioneClassifica)

FK: Conclusa[Tappa, CodiceCiclista, SquadraCiclista] C Partecipazione[Tappa, CodiceCiclista, SquadraCiclista]

ModelloBici(Codice, Modello, Marca)

Tempo(Tappa, CodiceCiclista, SquadraCiclista, Valore)

FK: Tempo[Tappa, CodiceCiclista, SquadraCiclista] C Conclusa[Tappa, CodiceCiclista, SquadraCiclista]

Traduzione vincolo di generalizzazione completa

$\text{Cronometro}[Tappa] \cap \text{Normale}[Tappa] = \emptyset$

$\text{Tappa}[Codice] \subseteq \text{Cronometro}[Tappa] \cup \text{Normale}[Tappa]$

Traduzione vincolo su popolamento attributo tempo

$\forall x \in \text{istanze}(\text{Conclusa}) \exists y \in \text{istanze}(\text{Tempo}): y[\text{Tappa, CodiceCiclista, SquadraCiclista}] = x[\text{Tappa, CodiceCiclista, SquadraCiclista}]$ se e solo se $\exists z \in \text{istanze}(\text{Cronometro}) : x[\text{Tappa}] = z[\text{Tappa}]$

DOMANDA 3

Dato il seguente schema relazionale:

Fantino (Codice, Nominativo, Squadra)

Cavallo (Nome, Scuderia, Mantello, Sesso)

Gareggia (Corsa, CodiceFantino, NomeCavallo, Data)

Vince (Corsa, NomeCavallo, Data)

che contiene informazioni su fantini, cavalli, corse a cui partecipano e che vincono:

1) Restituire le corse in cui hanno gareggiato soltanto cavalli con sesso = 'F' (**3 Punti**)

```
SELECT DISTINCT Corsa FROM Gareggia WHERE Corsa NOT IN (SELECT
Corsa FROM Gareggia a, Cavallo b WHERE a.NoveCavallo = b.Nome
AND b.Sesso <> 'F')
```

2) Restituire i nomi dei cavalli che hanno gareggiato montati dal maggior numero di fantini diversi (**4 Punti**)

```
CREATE VIEW FantiniPerCavallo AS
SELECT NomeCavallo, count(distinct CodiceFantino) as Conteggio
FROM Gareggia
GROUP BY NomeCavallo
```

```
SELECT NomeCavallo FROM FantiniPerCavallo WHERE Conteggio =
(SELECT MAX(Conteggio) FROM FantiniPerCavallo)
```

3) Restituire il nome delle squadre che non hanno mai vinto una corsa (**4 Punti**)

```
SELECT DISTINCT Squadra FROM Fantino WHERE Squadra NOT IN
(SELECT c.Squadra FROM Vince a, Gareggia b, Fantino c WHERE
a.Corsa = b.Corsa AND a.NomeCavallo = b.NomeCavallo AND a.Data =
b.Data AND b.CodiceFantino = c.Codice)
```