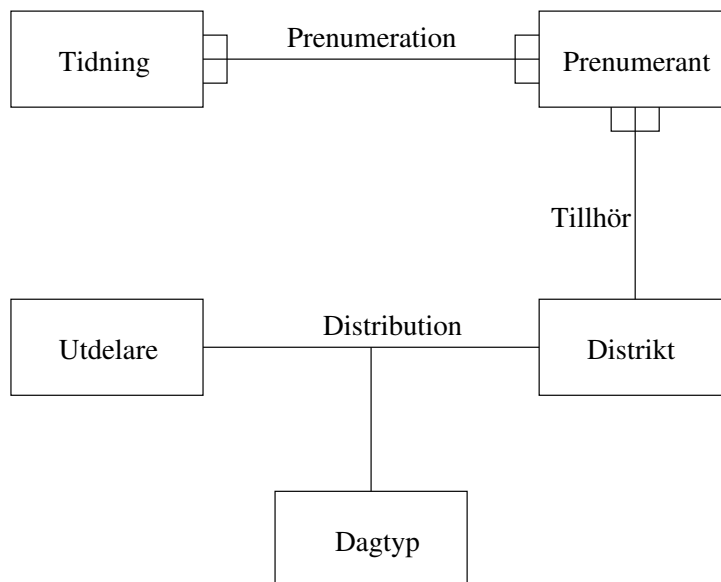


1. En databas över tidningsprenumerationer har tillverkats enligt nedanstående modell



med följande egenskapsmatris:

Typ	Namn	I-termer	E-termer
Obj	Tidning	TNamn	
	Prenumerant	Tel	PNamn, Adress
	Utdelare	Pnr	UNamn, Adress, Tel
	Distrikt	ID	Adress
	Dagtyp	Typ	
Samb	Prenumeration	Tnamn, tel	
	Tillhör	Tel, ID	
	Distribution	Pnr, Typ, ID	

där man använder telefonnumret till prenumeranten för att identifiera denne och distrikten har utlämningsställe för tidningarna på adressen "Adress".

- a) (5p) Överför modellen till databasstruktur. Motivera varje steg i överföringen (t.ex. genom angivande av vilka regler i "kokboken" som använts).
 - b) (5p) Man vill också tillåta att kunderna (prenumeranterna) får tidningen till annan adress under veckoslut. Föreslå en utbyggnad av modellen och egenskapsmatrisen så att även detta kan hanteras.
2. Tvärvetenskapliga projektarbeten på högskolan involverar oftast studenter från olika studieinriktningar. För att hålla reda på alla sådana projektarbeten och de studenter som arbetar med dem vill man ha en databas.

Varje projekt har naturligtvis ett start- och ett slutdatum. En institution ger kursen (projektarbetet). Man har behov av en historik över alla projekt, d.v.s. att man håller reda på såväl pågående som avslutade projekt.

Initialt har man samlat alla termer i en enda tabell.

- INamn Institutionens namn
- KNr Kursnummer
- KNamn Kursnamn
- Pnummer Projektets id
- Start Projektets startdatum
- Slut Projektets slutdatum
- Snr Students id
- SNamn Students namn
- SAdress Students adress
- SINamn Namn på den institution som studenten tillhör

a) (3p) Föreslå en bättre struktur. Du behöver inte genomföra en formell normalisering.

b) (2p) Varför kommer man att få problem om informationen lagras i ett enda register? (Specifika problem för den aktuella strukturen.)

3. Översätt till (begriplig?) svenska utgående från varuhuset, som har följande schema:

anställd	(<u>namn</u> , lön, chef, avd)
försäljning	(<u>avd</u> , <u>varunr</u> , volym)
leverantör	(<u>företag</u> , adress)
lager	(<u>företag</u> , <u>avd</u> , <u>varunr</u> , volym)
avdelning	(<u>avd</u> , våning)
vara	(<u>varunr</u> , typ)

a) (3p)

```
CREATE VIEW tmp(avd, n) AS
  SELECT avd, COUNT(*) FROM försäljning
  WHERE varunr in
    (SELECT varunr FROM vara WHERE typ='cykel')
  GROUP BY avd;
```

```
CREATE VIEW max AS
  SELECT MAX(n) FROM tmp
```

```
SELECT tmp.avd FROM tmp, max WHERE tmp.n = max.n;
```

b) (2p)

```
SELECT avd FROM försäljning F WHERE EXISTS
  (SELECT * FROM försäljning WHERE F.avd <> avd
    AND F.varunr = varunr);
```