

Tentamen i EDAF75

14 mars 2023

Skrivtid: 14-19

- SKRIV BARA PÅ ENA SIDAN AV PAPPRET – tentorna kommer att scannas in, och endast framsidorna rättas.
- SKRIV TYDLIGT – om texten inte går att läsa kan du inte få några poäng.
- SÄTT IDENTITET OCH SIDNUMMER PÅ VARJE INLÄMNAT BLAD, kontrollera noga att du lämnar in alla blad, och lägg dem i rätt ordning i omslaget.
- *Preliminär* maxpoäng anges vid varje uppgift.
- *Preliminära* betygsgränser:
 - 3: Minst ca 60% på *var och en* av uppgift 1-3 (alla tre måste vara godkända).
 - 4: Minst ca 70% på *var och en* av uppgift 1-3, och ca 70% totalt på uppgift 4-6.
 - 5: Minst ca 85% på *var och en* av uppgift 1-3, och ca 85% totalt på uppgift 4-6.

Uppgift 1

I denna uppgift förekommer ett antal termer som kan vara svåra att översätta till engelska under tentan, om du vill får du gärna använda svenska namn i din modell (å, ä och ö går bra!).

I en botanisk undersökning undersöker man var olika växter förekommer, i hela Sverige. Växterna identifieras med sina latinska namn: *leucanthemum vulgare*, *hedera helix*, etc. Undersökningarna äger rum på olika platser, vi kallar dem "lokaler" – en lokal beskrivs med sitt namn (till exempel "Dalby Söderskog"), sin typ ("Sydsvensk lövskog") och GPS-koordinaterna för dess mittpunkt (55.676283 N, 13.328820 E). Vi kan anta att lokalernas namn är unika.

Inom en lokal sker inventeringen i *rutor* med storleken 1×1 m. Varje ruta har en egen mittpunktskoordinat, och koordinaterna mäts i meter, relativt GPS-koordinaterna för lokalens mittpunkt.

För alla växter som förekommer i en ruta noteras hur många procent av rutan som täcks av växten vid ett givet tillfälle.

Det görs också kemiska analyser av miljön i en del av rutorna, exakt vilka analyser som görs kan variera, men vanligt är att man mäter pH och halten av olika tungmetaller, och man gör ofta samma analys i rutan vid olika tidpunkter. Resultatet av mätningarna är alltid numeriska värden, och ges i olika enheter, som pH, ppm, g,

Vid inventeringen är ett stort antal personer inblandade, för varje person håller man reda på personnummer, namn och adress. Varje ruta inventeras av en given person.

- (a) Gör en E/R-modell som beskriver databasen, redovisa med ett UML-diagram, markera nycklar, och skriv ut multipliciteter på samtliga associationer. *Skriv inte ut främmande nycklar som redan finns markerade i diagrammet.* 8 p
- (b) Skriv SQL-satser för att definiera samtliga tabeller för din E/R-modell – var noga med att deklarerar primärnycklar och främmande nycklar. Du får gärna använda samma svenska namn som i (a) på tabeller och attribut. 3 p
- (c) Utgående från dina relationer i uppgift b, skriv en SQL-sats som ger namnen på alla växter som förekommer i områden med typen "Sydsvensk lövskog". 2 p

Uppgift 2

Ett antal affärer säljer varor (items) till olika priser, och kunder köper varor från affärerna. Detta beskrivs av följande relationer (understrukna attribut är delar av primärnycklar, kursiverade fetstilta attribut är främmande nycklar):

```
shops(shop_id, name, address)
items(item_id, name, description)
offers(shop_id, item_id, price)
customers(customer_id, name, address)
sales(sales_id, customer_id, item_id, shop_id, sales_date)
```

Rita ett diagram i uppgift (a), och skriv SQL-satser för övriga uppgifter:

- (a) Rita ett ER-diagram i UML-format för att beskriva databasen, rita diagrammet enligt anvisningarna i problem 1a. *2 p*
- (b) Skriv ut namn och adress för alla kunder som inte köpt någon vara under 2022. TIPS: man kan få året för ett datum med hjälp av standardfunktionen year. *2 p*
- (c) Skriv ut antalet affärer som säljer varor med ett item_id som börjar på 'EF'. *3 p*
- (d) Skriv ut namn och adress för den affär (eller de affärer) som säljer varan med item_id = 'EF123-A' till lägst pris. *3 p*
- (e) För alla kunder som handlat för minst 1 000 kronor: skriv ut kund-id, namn och totalsumman för alla inköp, sortera efter minskande totalsumma. *3 p*

Uppgift 3

För att ingen skall tappa onödiga poäng på ett eventuellt misstag i (a), så har vi helt separerat uppgift (a) från uppgift (b)..(e).

- (a) I en kurs skall studenterna göra 3 laborationer i grupper om 2-3 studenter – för att studenterna skall få övning i att arbeta med olika personer byts lab-grupperna för varje laboration, så en student kan till exempel arbeta i grupp 12 på laboration 1, och grupp 29 på laboration 2.

Vi har följande attribut:

- `stil_id`: student-id
- `name`: namn på student (inte nödvändigtvis unikt)
- `lab_number`: laborationsnummer (1-3)
- `group_number`: grupp-nummer (1-50) för en given laboration
- `status`: status för en grupp för en given laboration (ändras när gruppen blir godkänd)

Vi antar att alla studenter har fått en grupp att jobba i för samtliga laborationer, och att alla studenter i en grupp blir godkända samtidigt på en laboration (dvs vi antar att ingen missar redovisningen).

Bestäm de funktionella beroenden som gäller för attributen ovan, och *använd beteckningarna ovan* (dvs introducera *inte* nya namn som A, B, etc.).

4 p

- (b) I relationen $R(A, B, C, D, E, F)$ gäller följande funktionella beroenden:

$$FD_1: A \rightarrow CF$$

$$FD_2: C \rightarrow A$$

$$FD_3: CE \rightarrow B$$

$$FD_4: F \rightarrow D$$

Bestäm relationens nycklar, motivera ditt svar.

2 p

- (c) Visa att relationen inte är i BCNF.

1 p

- (d) Bryt ner relationen R i (b) i mindre relationer som är i BCNF, och som kan rekonstrueras till den ursprungliga relationen utan att vi förlorar eller lägger till rader. Motivera varje steg i nedbrytningen genom att skriva dina relationer på nedanstående form, och förklara de uppdelningar du gör:

$R(A, B, C, D, E, \dots)$
Beroenden: $A \rightarrow BC, BD \rightarrow E, \dots$
Nycklar: AC, AD, \dots

Att svara genom att bara ange relationer utan motivering ger inga poäng.

5 p

- (e) Visa med en SQL-sats hur vi kan återskapa den ursprungliga relationen i (b) med hjälp av relationerna som du bröt upp den i ovan.

1 p

Uppgift 4

Definiera en trigger som gör att ingen kund i databasen från uppgift 2 kan handla för mer än 32 000 kronor under en dag.

4 p

Uppgift 5

Skriv ett Python- eller Javaprogram som med databasen i uppgift 2 löser följande uppgift:

Läs in ett artikelnummer (item_id), och skriv ut de (högst) tre dagar då varan såldes flest gånger – skriv även ut antalet gånger den såldes de aktuella dagarna.

Databasen ligger i SQLite-filen "shopping.sqlite".

Du skall skriva ett program som går att köra, men du behöver inte skriva några import-satser. Om du skriver Javakod måste du använda JDBC-paketet, och om du skriver Pythonkod måste du använda sqlite3-biblioteket.

För att läsa in en sträng kan du använda följande anrop:

- I Java: `var s = EDAF75.nextLine("Ange s: ");`
- I Python: `s = input("Ange s: ")`

4 p

Uppgift 6

(a) Förklara vad bokstäverna i akronymen ACID betyder, och hur dessa begrepp relaterar till transaktioner.

1.5 p

(b) Beskriv kortfattat var och en av de fyra isolationsnivåer för transaktioner som beskrivs i kursen.

1.5 p