

Örebro universitet
Institutionen för naturvetenskap och teknik
[Thomas Padron-McCarthy \(thomas.padron-mccarthy@oru.se\)](mailto:thomas.padron-mccarthy@oru.se)

Tentamen i

Databasteknik

för D1, SDU1 m fl

lördag 7 mars 2015

Gäller som tentamen för:

DT1026 Datateknik A, Databasteknik, provkod 0100
DT1030 Datateknik A, Tillämpad datavetenskap, provkod 0310

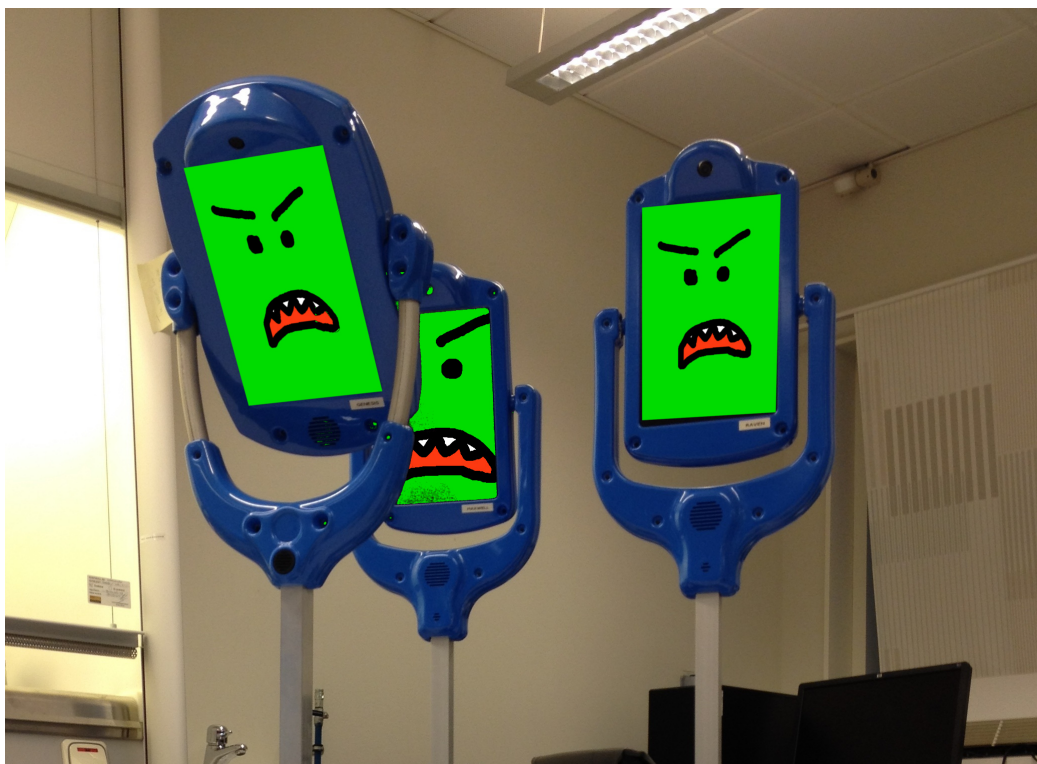
Hjälpmedel:	Inga hjälpmedel.
Poängkrav:	Maximal poäng är 34. För betyget 3 respektive G krävs 17 poäng.
Resultat:	Meddelas på kursens hemsida eller via e-post senast lördag 28 mars 2015.
Återlämning av tentor:	Efter att resultatet meddelats kan tentorna hämtas på universitetets centrala tentamensutlämning.
Examinator och jourhavande:	Thomas Padron-McCarthy, telefon 070 - 73 47 013. Det är möjligt att läraren inte alltid går att nå.

- Skriv tydligt och klart. Lösningar som inte går att läsa kan naturligtvis inte ge några poäng. Oklara formuleringar kommer att misstolkas.
 - Skriv den personliga tentamenskoden på varje inlämnat blad. Skriv *inte* namn eller personnummer på bladen.
 - Skriv bara på en sida av papperet. Använd inte röd skrift.
 - Antaganden utöver de som står i uppgifterna måste anges. Gjorda antaganden får inte förändra den givna uppgiften.
 - Skriv gärna förklaringar om hur du tänkt. Även ett svar som är fel kan ge poäng, om det finns med en förklaring som visar att huvudtankarna var rätt.
-

LYCKA TILL!

Scenario till uppgifterna

Forskningsgruppen AASS vid Örebro universitet forskar om robotar. De påstår att robotarna ska användas i gruvor och på äldreboenden, men egentligen är det förstås onda stridsrobotar man bygger.



Nu behöver de en databas för att hålla reda på robotarna.

Det vi ska lagra i databasen är följande:

- **Robottyper.** Varje robottyp har ett unikt namn, till exempel **Strids-giraff 2000**, och ett unikt nummer.
- **Robotar.** Varje robot har ett unikt nummer och är av en viss robottyp. Dessutom har den ett aktiveringsdatum.
- **Strider.** I varje strid slåss två *eller flera* robotar mot varandra. Varje strid sker ett visst datum, men det kan ske flera strider samma dag. Dessutom kan det i varje strid finnas ett antal vinnare och förlorare. Av de robotar som deltagit i striden kan var och en vara vinnare eller förlorare, eller ingetdera. I varje strid kan det alltså finnas noll eller flera vinnare och noll eller flera förlorare. En robot kan inte vara både vinnare och förlorare i samma strid. En robot kan delta i hur många strider som helst, och vinna hur många strider som helst, men när den förlorat en strid är den förstörd och kan inte delta i fler strider.
- **Skador.** Robotarnas strider orsakar skador på universitetet. Varje skada hör ihop med en viss strid, och den har också en beskrivning och en kostnad.

Uppgift 1 (5 p)

Rita ett ER- eller EER-diagram för den beskrivna databasen. Använd informationen i scenariot ovan, men tänk också på att det ska gå att svara på frågorna i uppgift 3 nedan.

ER- och EER-diagram kan ritas på flera olika sätt. Om du använder en annan notation än kursboken, måste du förklara den notation som du använder.

Uppgift 2 (5 p)

Implementera den beskrivna databasen i relationsmodellen, dvs översätt ER-diagrammet till tabeller.

Du behöver inte skriva **create table**-kommandon i SQL, men du ska ange vilka relationer som finns och vilka attribut varje relation innehåller. Ange också alla kandidatnycklar, vilken av dessa som är primärnyckel, samt vilka referensattribut som finns och vad de refererar till.

Implementationen ska vara bra.

Uppgift 3 (10 p)

Formulera följande frågor i SQL. Använd dina tabeller från uppgift 2. Definiera gärna vyer om det underlättar.

- a) Här är robot nummer **826**. Vad heter robottypen?
- b) Hur många robotar har vi?
- c) Hur många robotar har vi som inte är förstörda? En robot är förstörd om den förlorat en strid.
- d) Det står i scenariot att en robot inte kan vara både vinnare och förlorare i samma strid. Kanske har vi ändå råkat lägga in en robot som vinnare och förlorare i samma strid? Skriv en SQL-fråga som visar vilka robotar och strider det i så fall är!

(Alternativt: Om din databas från uppgift 2 gör att man inte *kan* göra detta fel, förklara varför det inte går!)

- e) Vilken dag inträffade den dyraste striden, dvs den strid som gav störst sammanlagd kostnad för skadorna den orsakade?

Uppgift 4 (3 p)

SQL-frågorna a, b och c i uppgiften ovan kommer att köras ofta, och vi vill att de ska gå snabbt att köra. Vilka index behöver vi skapa för att snabba upp just dessa frågor? Databashanteraren som vi använder skapar inga index automatiskt, inte ens på primärnycklar.

Allteftersom robotar blir billigare, siktar AASS på att köpa in tusentals och kanske miljoner robotar, av många olika typer. De har förstås strider hela tiden.

Uppgift 5 (5 p)

Medan de väntade på att du skulle bli klar med uppgift 2 gjorde forskarna på AASS en egen tabell, tabellen **Robotar**, så att de nödtorftigt kunde lagra data om sina robotar:

Robotnr	Robotypnr	Robotyp	Aktiveringsdatum
826	71	Strids-giraff 2000	2012-03-22
827	72	Attackrullare X-1	2012-03-22
828	72	Attackrullare X-1	2012-04-02

a) Vilka kandidatnycklar finns i tabellen?

b) Vilka fullständiga funktionella beroende finns i tabellen?

c) Vilken är den högsta normalform som tabellen uppfyller, av 1NF, 2NF, 3NF och BCNF? (Ledtråd: Det är inte BCNF.) Motivera svaret!

d) Beskriv ett problem som finns i den här tabellen, men som man skulle slippa om den hade uppfyllt BCNF.

Uppgift 6 (3 p)

Ange tre olika databashanterare. Skriv för var och en åtminstone någon egenskap som skiljer den från de andra.

Uppgift 7 (3 p)

Vi använder en databashanterare som har så kallad "auto-commit", vilket betyder att varje SQL-kommando räknas som en egen transaktion, så länge man inte uttryckligen startar en transaktion med kommandot "start transaction".

Vi startar två olika klientprogram som loggar in på samma databas, och ger följande SQL-kommandon, i den angivna ordningen, i de två klienterna;

Klient 1	Klient 2
create table Strutsar (nummer integer not null primary key, status integer);	
	insert into Strutsar values (1, 10);
select * from Strutsar; -- <i>Fråga 1</i>	
insert into Strutsar values (2, 20);	
	select * from Strutsar; -- <i>Fråga 2</i>
	start transaction; insert into Strutsar values (3, 30); rollback;
select * from Strutsar; -- <i>Fråga 3</i>	
start transaction; insert into Strutsar values (4, 40);	
select * from Strutsar; -- <i>Fråga 4</i>	
	select * from Strutsar; -- <i>Fråga 5</i>
commit;	
	select * from Strutsar; -- <i>Fråga 6</i>
select * from Strutsar; -- <i>Fråga 7</i>	

Vad blir resultatet av var och en av de sju select-frågorna?
