

Het examen is volledig open boek. Er mag gebruik gemaakt worden van al het materiaal voorzien tijdens de lessen. Dit houdt in:

- Het boek 'Principes van databases' van prof. dr. Guy De Tré
- Alle slides, opgaves, (opgeloste) oefeningen, notities, examenvoorbereidingen,...

Het examen kan worden opgelost op lege witte bladeren. **Los elk deel van het examen (sectie 1-4) op op een apart blad.** Vermeld zeker op **elk** oplossingsblad je studentenummer, naam en richting.

Opgelet: omwille van de coronamaatregelen kunnen wij tijdens het examen geen vragen beantwoorden. Indien er vragen of onduidelijkheden zijn, schrijf deze dan nauwgezet op je oplossingsblad zodat wij hier tijdens het quoteren rekening mee kunnen houden.

## 1 Open vragen

Beantwoord onderstaande vragen.

1. Hoe kan je bij het gebruik van null-waarden onvolledige query-resultaten voorkomen?
2. Geef aan welke van de onderstaande uitspraken **fout** zijn en motiveer waarom deze fout zijn.
  - a) Een cursor kan worden gebruikt voor het verwijderen van data.
  - b) Na gebruik moet een cursor steeds expliciet worden afgesloten.
  - c) Met de `declare`-instructie geef je aan dat de opgegeven `select`-instructie moet worden uitgevoerd zodat de tuples uit het resultaat verder kunnen worden verwerkt.
  - d) Bij Embedded SQL moet je een cursor gebruiken om een `select`-instructie met meerdere resultaten correct te kunnen verwerken.
  - e) Een cursor kan binnen een programma meerdere keren worden geopend.
3. Waarom werken optimistische methoden enkel efficiënt als vrijwel alle transacties slagen voor de validatietest?
4. Welke NoSQL-systemen zijn de beste oplossing voor het snelheidsprobleem? Verklaar kort waarom.

## 2 Conceptueel ontwerp

De vijf Vlaamse provincies startten enkele jaren terug met het ontwikkelen van een heus **wandelknooppuntennetwerk** in Vlaanderen. Het idee hierachter is dat mensen met behulp van deze netwerken zelf hun route langs verschillende knooppunten heen kunnen samenstellen, in plaats van enkel maar de mogelijkheid te hebben om vooropgestelde routes te wandelen. Om alle informatie omtrent deze wandelnetwerken gestructureerd te kunnen opslaan in een databank, is er uiteraard in eerste instantie een conceptueel ontwerp nodig. Enkele databankspecialisten ontwierpen zo een conceptueel ontwerp op basis van onderstaande **informatie die rechtstreeks uit de informatievergaringsfase komt. Dit conceptueel ontwerp kan je vinden in Appendix 1.**

De verschillende provincies die deelnemen aan het wandelknooppuntennetwerk kunnen uniek worden geïdentificeerd door hun naam. Ook het adres van de hoofdzetel van de provincie wordt opgeslagen. Een provincie is verantwoordelijk voor één of meerdere wandelnetwerken, die op hun beurt ook een unieke naam hebben. Deze wandelnetwerken bevatten wandelknooppunten, met een knooppuntnummer dat uniek is binnen een wandelnetwerk. Daarnaast heeft ieder wandelknooppunt ook een naam en een bijhorende beschrijving. Op die manier hoort ieder wandelknooppunt eenduidig bij een specifieke provincie.

Tussen twee verschillende wandelknooppunten kan (maximaal) één bewegwijzerde route aanwezig zijn. Een route kan enkel aanwezig zijn tussen twee wandelknooppunten die tot hetzelfde wandelnetwerk behoren.

Tenslotte dient in de databank te worden opgeslagen welke wandelinstructies er nodig zijn (en in welke volgorde ze moeten worden uitgevoerd) om via de bewegwijzerde route van het eerste naar het tweede wandelknooppunt te geraken (bv. “sla linksaf aan de kerk van Zottegem” of “Loop rechtdoor tot u aan huisnummer 53 komt”).

Beantwoord nu onderstaande vragen.

1. In het EER-diagram dat de ontwerpers opstelden zitten één of meer syntactische fouten. Dit zijn fouten die indruisen tegen de regels van hoe een correct EER-diagram er moet uitzien en zijn dus geen fouten tegen de semantiek van het probleem waarvoor een ontwerp wordt opgesteld. Pas het EER-diagram in Appendix 1 aan zodat er geen fouten tegen de EER-syntax (zoals gezien in de oefeningenlessen) meer aanwezig zijn. Het is dus niet de bedoeling om semantische fouten te verbeteren, maar zorg ervoor dat je bij het verbeteren van de syntactische fouten geen (extra) semantische fouten introduceert in het schema. Hou dus rekening met de informatie uit de informatievergaringsfase. Je mag je aanpassingen rechtstreeks op het EER-diagram in Appendix 1

maken. Zorg wel dat je Appendix 1 zeker mee indient met je antwoordenblad!

2. In het EER-diagram (zie Appendix 1) zijn twee cycli aanwezig. Zijn deze cycli noodzakelijk of kan het schema aangepast worden zodat er minder (of zelfs geen) cycli overblijven, maar de semantiek behouden blijft? Om welke aanpassing(en) zou het dan gaan? Bespreek bovenstaande vragen voor beide cycli afzonderlijk.
3. In het conceptueel ontwerp wordt een wandelinstructie gemodelleerd als een apart entiteitstype dat uniek wordt geïdentificeerd door een willekeurig id (surrogaatsleutel) en dat daarnaast enkel nog de effectieve beschrijving van de wandelinstructie bevat (bv. "linksaf"). Zorgt de huidige manier van modellering van wandelinstructies voor een eenduidige wegbeschrijving van de route tussen twee wandelknooppunten of ontbreekt er op dit moment nog informatie in het diagram om exact te kunnen bepalen hoe je van wandelknooppunt A naar wandelknooppunt B stapt? Zo ja, welke informatie ontbreekt er nog en hoe zou je het huidige conceptueel ontwerp aanpassen zodat deze informatie correct kan worden gemodelleerd. Bespreek bovenstaande vragen. Je hoeft geen rechtstreekse aanpassingen op de figuur in Appendix 1 te maken, het is genoeg om in woorden je antwoord te schetsen.
4. Stel dat gebruikers (uniek geïdentificeerd door een username) zelf wandelingen, uniek geïdentificeerd door een naam, zouden kunnen voorstellen en opslaan die lopen over meerdere opeenvolgende bewegwijzerde routes. Wandelingen moeten beginnen en eindigen op exact hetzelfde wandelknooppunt, zodat ze een lus vormen. Hoe zou je dit modelleren in het conceptueel ontwerp? Teken op je antwoordblad hoe je dit zou aanpakken. Je hoeft niet het hele schema uit Appendix 1 opnieuw te tekenen, maar maak duidelijk waar jouw aanpassingen zich zouden bevinden ten opzichte van de rest van het schema. Maak ook duidelijk of er aanpassingen aan de functionele beschrijving moeten gebeuren.

### 3 Logisch ontwerp

Verklaar, in je eigen woorden, hoe je een *disjuncte* overerving uit het conceptueel ontwerp volledig en correct omzet bij de mapping naar het logisch ontwerp. Let op, het is niet de bedoeling om te bespreken hoe je een overerving omzet, wel hoe je ervoor zorgt dat de overerving *disjunct* blijft.

## 4 Workshops

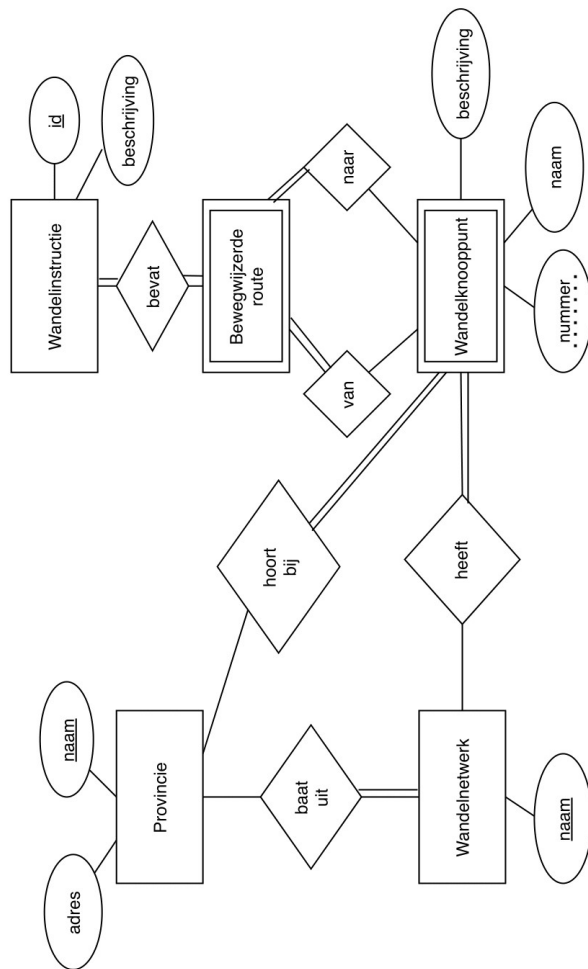
In Appendix 2 is de DDL-code gegeven van een databank voor de opslag van agenda-items. Bekijk deze DDL-code nauwgezet en los daarna onderstaande vragen op.

1. Waar/niet waar. Het is niet mogelijk om, bij de aanmaak van een agenda, een NULL-waarde op te geven voor de naam van de agenda. Verklaar je antwoord.
2. Waar/niet waar. De naam van een agenda is voldoende om de kleur van deze agenda uniek te bepalen. Verklaar je antwoord.
3. Waar/niet waar. Bij het verwijderen van een agenda worden automatisch ook alle evenementen, gelinkt aan deze agenda, en alle genodigden (dus niet gebruikers!), gelinkt aan deze evenementen, verwijderd uit de databank. Verklaar je antwoord.
4. Stel dat de databank reeds gevuld is met data, zoals weergegeven in Appendix 3. Hoe zal de resultatentabel na uitvoering van onderstaande query eruitzien. Teken deze tabel *volledig*.

```
SELECT * FROM view1;
```

5. Stel dat je een trigger wil implementeren die ervoor zorgt dat gebruikers niet uitgenodigd kunnen worden op de evenementen die gelinkt zijn aan hun eigen agenda's.
  - Op welke tabel zal de trigger gedefinieerd moeten worden?
  - Beschrijf stap voor stap *in woorden* (dus niet met code!) de logica die moet worden geïmplementeerd in de functie gekoppeld aan deze trigger om ervoor te zorgen dat bovenstaande vereiste vervuld is.

## Appendix 1: Conceptueel ontwerp wandelknooppuntennetwerk



### Functionele beschrijving

- wandelknooppunt.nummer > 0
- bewegwijzerde routes mogen enkel wandelknooppunten verbinden die tot hetzelfde wandelnetwerk behoren.

## Appendix 2: DDL-code agendadatabank

```
CREATE TABLE gebruiker
(
    email          VARCHAR PRIMARY KEY,
    voornaam       VARCHAR NOT NULL,
    achternaam     VARCHAR NOT NULL
);

CREATE TABLE agenda
(
    email VARCHAR REFERENCES gebruiker (email),
    naam  VARCHAR,
    kleur VARCHAR(1) CHECK ( kleur IN ('r', 'g', 'b') ) NOT NULL,
    CONSTRAINT agenda_pk PRIMARY KEY (email, naam)
);

CREATE TABLE evenement
(
    id          SERIAL      PRIMARY KEY,
    naam        VARCHAR     NOT NULL,
    begintijd   TIMESTAMP   NOT NULL,
    duur        INTERVAL    NOT NULL,
    agenda_naam VARCHAR     NOT NULL,
    agenda_email VARCHAR     NOT NULL,
    CONSTRAINT evenement_fk FOREIGN KEY (agenda_naam, agenda_email)
        REFERENCES agenda (naam, email),
    CONSTRAINT duur_check CHECK ( duur > '0 seconds' )
);

CREATE TABLE genodigden
(
    evenement_id INTEGER REFERENCES evenement (id)      NOT NULL,
    email         VARCHAR REFERENCES gebruiker (email) NOT NULL,
    CONSTRAINT genodigden_pk PRIMARY KEY (evenement_id, email)
);

CREATE VIEW view1 AS
    SELECT a.id, a.email FROM
        (SELECT email, id FROM gebruiker, evenement
        EXCEPT
        SELECT email, evenement_id FROM genodigden) a
    INNER JOIN evenement ev ON a.id = ev.id
    WHERE ev.begintijd - now() BETWEEN '0 days'::INTERVAL AND '7 days'::INTERVAL;

-- je mag veronderstellen dat now() een timestamp teruggeeft die
-- overeenkomt met het begin van dit examen (namelijk 5 januari 2021, 08u30)
```

### Appendix 3: Voorbeelddata agendadatabank

<b>email</b>	<b>voornaam</b>	<b>achternaam</b>
jan.janssens@ugent.be	Jan	Janssens
piet.pieters@ugent.be	Piet	Pieters
tim.timmers@ugent.be	Tim	Timmers

Tabel 1: gebruiker

<b>email</b>	<b>naam</b>	<b>kleur</b>
jan.janssens@ugent.be	Unief	r
jan.janssens@ugent.be	Vrije tijd	g
piet.pieters@ugent.be	Sport	b

Tabel 2: agenda

<b>id</b>	<b>naam</b>	<b>begintijd</b>	<b>duur</b>	<b>agenda_naam</b>	<b>agenda_email</b>
1	Examen db	5 januari 2021, 8u30	3 uur	Unief	jan.janssens@ugent.be
2	Studeren	17 december 2020, 13u	5 uur	Unief	jan.janssens@ugent.be
3	Kerstfeest	24 december 2020, 19u	5 uur	Vrije tijd	jan.janssens@ugent.be
4	Lopen	9 januari 2021, 19u	1 uur	Sport	piet.pieters@ugent.be
5	Thesis	15 januari 2021, 9u	2 uur	Unief	jan.janssens@ugent.be

Tabel 3: evenement

<b>evenement_id</b>	<b>email</b>
1	piet.pieters@ugent.be
1	tim.timmers@ugent.be
3	jan.janssens@ugent.be
3	piet.pieters@ugent.be
4	jan.janssens@ugent.be
4	piet.pieters@ugent.be

Tabel 4: genodigden