

EVALUACIÓN	Solución del examen	GRUPO	Todos	FECHA	17/Abr/2015
MATERIA	Bases de datos y Bases de datos 1				
CARRERA	AP – ATI				
CONDICIONES	- Puntos: 100 - Duración: 2 1/2 horas - SIN material				

Ejercicio 1

En un instituto de enseñanza se registra información en planillas electrónicas y ahora se desean volcar en un sistema informático. Se guarda información sobre los estudiantes como ser la cédula de identidad (no todos los estudiantes cuentan con ésta), el nombre donde debe distinguirse los nombres de los apellidos para poder ordenar las listas entre otras cosas, la fecha de nacimiento, una dirección detallada (calle, número de puerta con o sin bis, apto, departamento, ciudad, etc), teléfonos de contacto (pueden ser fijos o celulares) y direcciones de correo electrónico que utiliza habitualmente. A cada uno de ellos se les asigna un número de estudiante único que los identifica en la institución.

Cada estudiante que se inscribe en el instituto puede elegir entre todos los cursos que se dictan. De cada curso se conoce un código que lo identifica, el nombre (todos lo tienen y no se repite), la duración (en cantidad de horas “frontales”) y la cantidad de horas por semana. También se conoce si para realizar un curso se debe haber aprobado otro previamente.

Otra dato que se registra es los docentes del instituto. De cada uno de ellos se conoce el número de funcionario que los identifica, la cédula de identidad uruguaya (obligatoria por el BPS), el nombre, la dirección, el número de teléfono celular personal y una dirección de correo brindado por la institución que también es obligatorio y tampoco se repite.

Se sabe que docentes pueden dictar que cursos. Puede que algunos docentes nunca hayan dictado algún curso de los que está capacitado pero igual está registrado en el sistema para poder ofrecerles cubrir alguna ausencia de los profesores que tienen asignado el dictado.

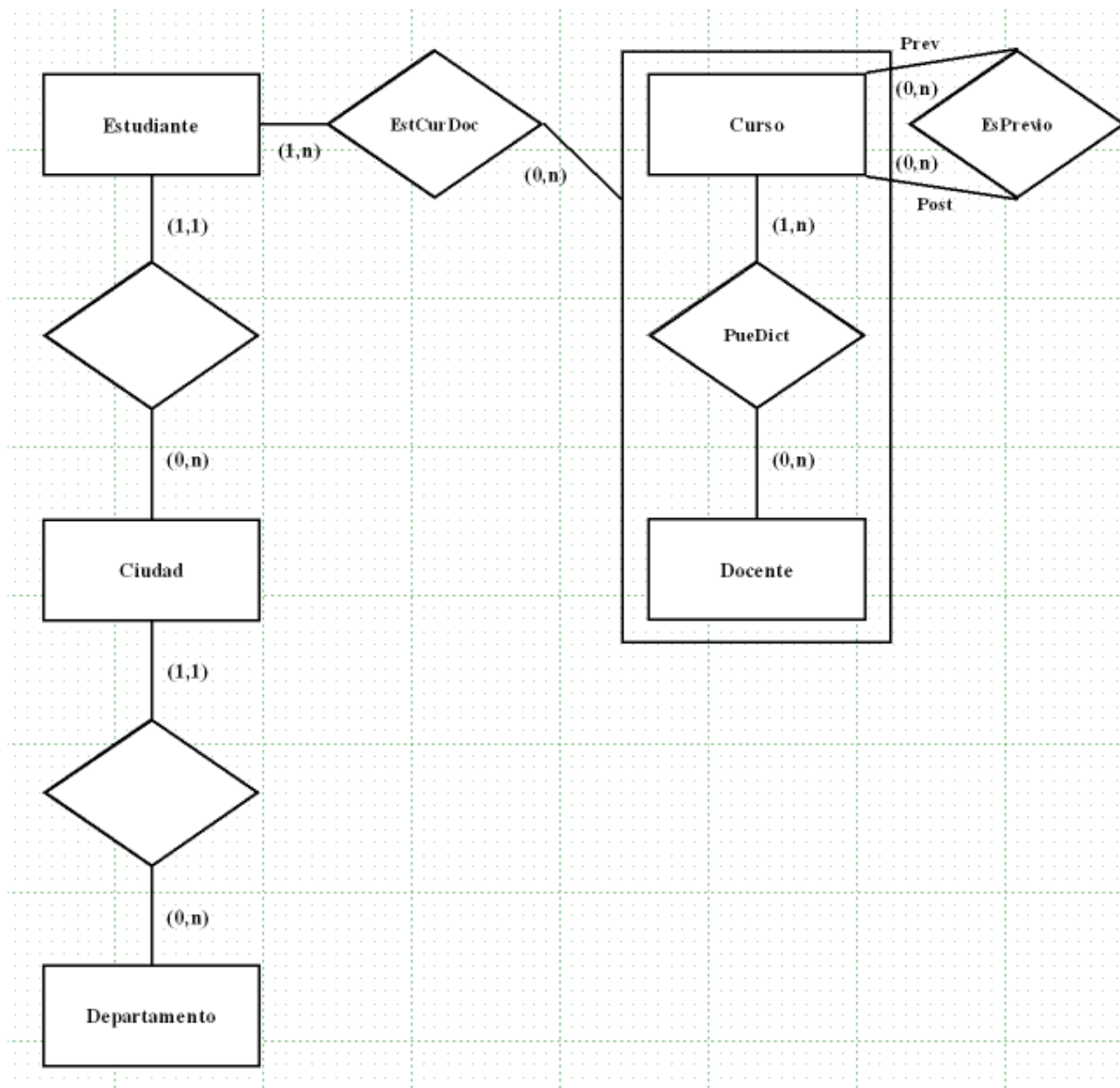
De cada estudiante que ha finalizado un curso se sabe la cantidad de faltas en horas y una nota final que indicará si aprobó el curso al comparar estos valores con la nota mínima necesaria para aprobar la materia y el porcentaje máximo de ausencias admitidas.

Se debe tener en cuenta que el sistema debe prever que se sepa con qué docente (o docentes si recurrió) realizó un curso cada estudiante. Al momento de inscribirse al curso ya queda definido con cuál de los docentes que pueden dictarlo lo realizará.

También se debe tener en cuenta que puede haber profesores que dejen de dictar un curso (una vez finalizado) pero debe conservarse la información de los alumnos que tuvo en cada curso que haya dictado.

- Se pide modelo entidad-relación correspondiente donde se debe especificar claramente los atributos de cada entidad y relación y en caso de que no sean obvios, que representen. También se debe especificar la cardinalidad y participación de las entidades en las relaciones y en caso de utilizar este tipo de estructuras, las agregaciones y/o categorizaciones/generalizaciones, entidades débiles y evitar toda situación que pudiera permitir el almacenamiento de datos inconsistentes que no se pueda deducir del modelo utilizando restricciones de integridad no estructurales. **(Máximo: 30 puntos)**
- Convertir el MER anterior en su correspondiente esquema relacional llevándolo al menos hasta 3ª forma normal y explicitando sus correspondientes restricciones de integridad (estructurales y no estructurales). **(Máximo 20 puntos)**

Modelo Entidad-Relacion



Atributos de las Entidades:

Estudiante: EstCI₂, EstNro₁, EstNom, EstApe, EstNacFch, EstDirCalle, EstDirPuerta, EstDirNro, EstDirBis, EstDirApto, EstTel*, EstEmail*

Ciudad: CiuCod, CiuNom,

Departamento: DepCod, DepNom

Curso: CurCod₁, CurNom₂, CurDurac, CurCantHoraSem, CurNotaMinApr, CurPorcMaxAus

Docente: FunNro₁, DocCI₂, DocDir, DocEmail₃

Atributos de las Relaciones:

EstCurDoc: InscrFch, ComzoFch, Nota, CantFaltas

PueDict: DesdeFch, HastaFch

EsPrevio: No tiene atributos

Restricciones de Integridad (no estructurales):

1. Un curso no puede ser previo de si mismo ni directa ni transitivamente
2. Un estudiante no puede comenzar un curso si todavía no aprobó todos los previos
3. Un estudiante no se puede inscribir a un curso con un docente que ya no lo dicte
4. Un estudiante no puede tener mas faltas en un curso que la cantidad de horas totales del mismo
5. Un estudiante no se puede comenzar en la misma fecha el mismo curso con docentes diferentes

Modelo Relacional en 3ª Forma Normal:

Estudiante(EstCL₂, EstNro₁, EstNom, EstApe, EstNacFch, EstDirCalle, EstDirPuerta, EstDirNro, EstDirBis, EstDirApto, CiuCod) CiuCod es FK referencia a Ciudad

EsTel(EstNro, EstTel)

EstEmail (EstNro, EstEmail)

Ciudad(CiuCod, CiuNom, DepCod) DepCod es FK referencia a Departamento

Departamento(DepCod, DepNom)

Curso(CurCod₁, CurNom₂, CurDurac, CurCantHoraSem, CurNotaMinApr, CurPorcMaxAus)

Docente(FunNro₁, DocCL₂, DocDir, DocEmail₃)

PueDict(FunNro, CurCod, DesdeFch, HastaFch) FunNro es FK de Docente, CurCod es FK de Curso

EstCurDoc(EstNro, CurCod, ComzoFch, FunNro, InscrFch, Nota, CantFaltas)

{FunNro, CurCod} es FK de PueDict y EstNro es FK de Estudiante

EsPrevio(CurCodPrev, CurCodPost): Tanto CurCodPrev como CurCodPost son FKs de Curso

Restricciones de Integridad (no estructurales):

1. Un curso no puede ser previo de si mismo ni directa ni transitivamente
2. Un estudiante no puede comenzar un curso si todavía no aprobó todos los previos
3. Un estudiante no se puede inscribir a un curso con un docente que ya no lo dicte
4. Un estudiante no puede tener mas faltas en un curso que la cantidad de horas totales del mismo

Ejercicio 2

Dado el siguiente conjunto de relaciones, escribir las consultas en Álgebra relacional, que respondan a los requerimientos enunciados:

País (PaiCod, PaiNom)

Ciudad (CiuCod, CiuNom, PaiCod)

Aeropuerto (AepCod₁, AepNom₂, CiuCod, FchInaug, AepTipo)

Pista (AepCod, PisNro, PisLng)

Aerolínea (AelCod, AelNom, PaiCod, AerCat)

Escribir las sentencias SQL para las siguientes consultas teniendo en cuenta que:

- En todos los casos, se debe evitar datos repetidos si no aportan información útil para la consulta.
 - Se puede asumir concordancia en todas las claves foráneas pero no la participación total.
1. Obtener una lista de códigos y nombres de aerolínea que tengan bandera del mismo país que la aerolínea de código 'GW'.

```
SELECT AelCod, AelNom
FROM   Aerolinea U
WHERE  AelCod <> "GW"
AND    PaiCod = (SELECT D.PaiCod FROM Aerolinea D WHERE D.AelCod = "GW")
```

2. Obtener una lista de códigos y nombres de ciudades que no posean ningún aeropuerto internacional (tipo 'I').

```
SELECT CiuCod, CiuNom
FROM   Ciudad C
WHERE  CiuCod NOT IN(SELECT A.CiuCod
                     FROM   Aeropuerto A
                     WHERE   A.AepTipo = "I")
```

3. Obtener una lista de código ciudades del país de código 'ar' (Argentina) que tengan uno o más aeropuertos de cabotaje (tipo 'C') pero no tengan ningún aeropuerto internacional (tipo 'I').

```
SELECT CiuCod
FROM   Aeropuerto U
WHERE  PaiCod = "ar"
AND    AepTipo = "C"
AND    NOT EXISTS(SELECT D.*
                  FROM   Aeropuerto D
                  WHERE  U.CiuCod = D.CiuCod
                  AND    A.AepTipo = "I")
```

4. Obtener código y nombre del aeropuerto más nuevo de la ciudad de código 'snpb' (San Pablo) considerando que puede haber más de uno inaugurado en la misma fecha.

```
SELECT AepCod, AepNom
FROM   Aeropuerto U
WHERE  CiuCod = "snpb"
AND    FchInaug = (SELECT MAX(D.FchInaug)
                  FROM   Aeropuerto D
                  WHERE  D.CiuCod = "snpb")
```

5. Obtener una lista de nombres de aeropuertos que tengan más de 3 pistas de más de 2kms de longitud.

```
SELECT AepNom
FROM   Aeropuerto A, Pista P
WHERE  A.AepCod = P.AepCod
AND    PisLng > 2000
GROUP BY AepNom
HAVING (COUNT(*) >= 3)
```

(Cada consulta vale 10 puntos).