

FUNDAMENTOS DE BASES DE DATOS

Examen Diciembre 2023

La duración del examen es de 3 horas.

Presentar la resolución del examen:

- Con las hojas numeradas y escritas de un solo lado.
- Con la cantidad de hojas entregadas en la primer hoja.
- Con cédula de identidad y nombre en cada hoja.
- Escrita a lápiz y en forma prolija.
- Comenzando cada ejercicio en una nueva hoja

Ejercicio 1 (27 puntos)

La aerolínea Fing Airlines busca desarrollar un sistema para llevar un registro de sus operaciones de vuelo. De cada vuelo, se conoce un número que lo identifica, los servicios que ofrece a bordo (comida, entretenimiento, WiFi, etc.), su costo y sus distintos itinerarios. Cada itinerario de un vuelo se identifica por una fecha y se conoce su hora de salida y de llegada.

Cada itinerario opera en un aeropuerto que se identifica por su código IATA (International Air Transport Association) y de cada uno se conoce si es nacional o internacional. En cada aeropuerto se distribuyen puertas de embarque identificadas por un código dentro del mismo. De las puertas de embarque también se conoce la cantidad máxima de pasajeros que puede atender al mismo tiempo y su estado (disponible, en mantenimiento, fuera de servicio).

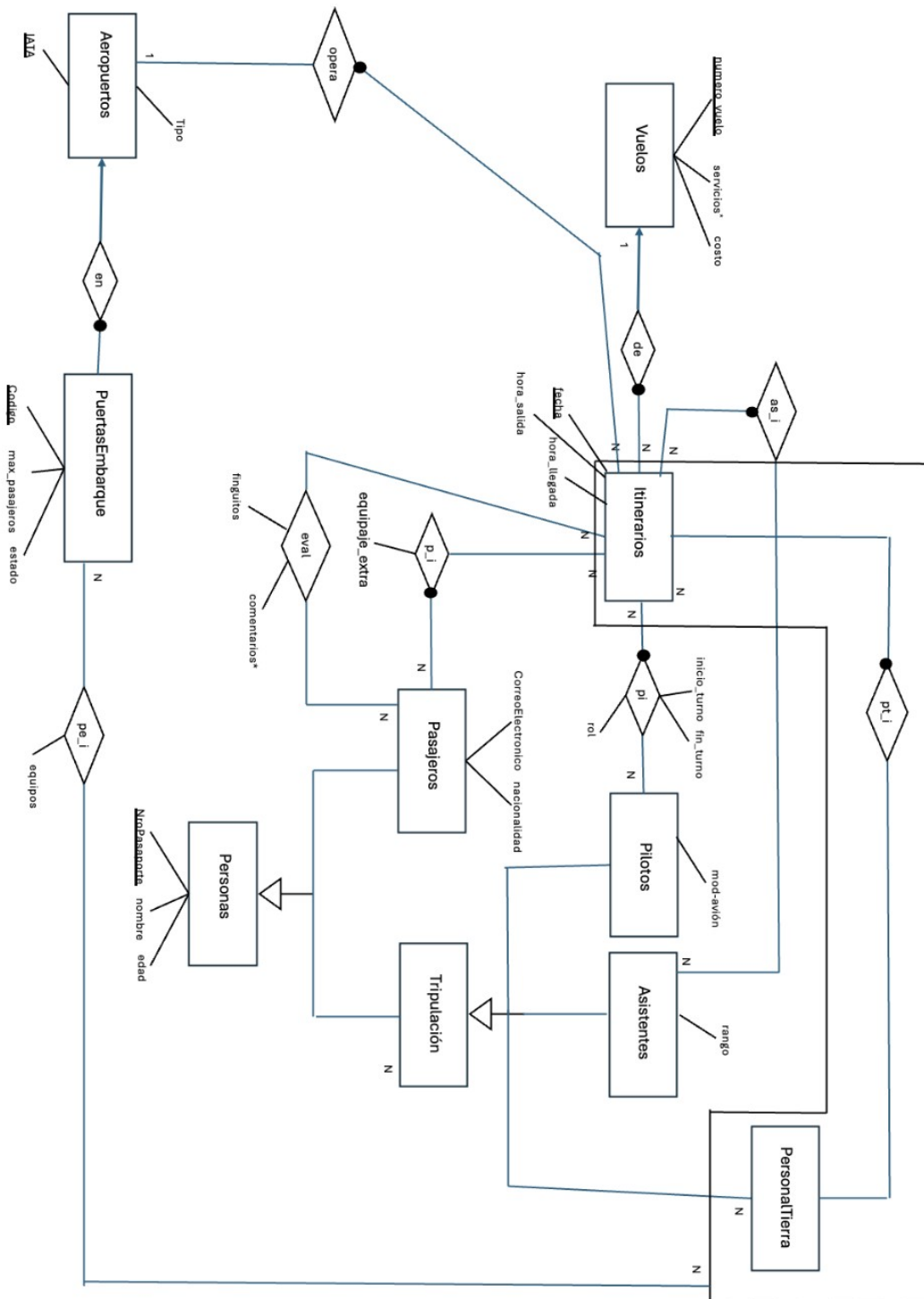
Las personas que se vinculan con los vuelos pueden ser pasajeros o tripulación. De todas las personas se conoce su número de pasaporte, nombre y edad. La tripulación se identifica con un número de empleado y además se conoce un teléfono de contacto. De los pasajeros se conoce un correo electrónico y es de interés saber su nacionalidad y para cada itinerario si viajaron con equipaje adicional.

La tripulación puede ser pilotos, asistentes de vuelo o personal de tierra. Se sabe que un miembro de la tripulación no puede cumplir más de una función. De los asistentes de vuelo se conoce su rango (senior, junior, etc.). Interesa registrar de todos los tripulantes en qué itinerarios trabajaron. Además, de los pilotos en cada itinerario interesa registrar el rol que tuvieron, así como el inicio y el fin de su turno. También interesa saber de cada piloto el modelo de avión que manejan.

Para cada miembro de la tripulación de tierra se conoce la puerta de embarque en la que se ubica para cada itinerario del que participa, así como también un conjunto de equipos con los que debe contar (radios, dispositivos de seguridad, etc.).

Los pasajeros pueden hacer evaluaciones a un itinerario de un vuelo. Estas tienen una puntuación asociada (denominados finguitos) y un conjunto de comentarios.

Se pide: Modelo Entidad-Relación completo del problema., incluyendo Restricciones No Estructurales.



Restricciones no estructurales

Pasajeros U Tripulación = Personas

Pilotos U Asistentes U PersonalTierra = Tripulación

Pilotos \cap Asistentes = \emptyset

Pilotos \cap PersonalTierra = \emptyset

AsistentesVuelo \cap PersonalTierra = \emptyset

Un itinerario debe de estar asociado a un piloto con rol capitán

$(\forall i \in ITINERARIOS) \exists (p \in PILOTOS) (\langle i, p \rangle \wedge rol \langle i, p \rangle = 'capitan')$

Un itinerario debe de estar asociado a un piloto con rol co-piloto

$(\forall i \in ITINERARIOS) \exists (p \in PILOTOS) (\langle i, p \rangle \wedge rol \langle i, p \rangle = 'co_piloto')$

Un pasajero solo puede evaluar vuelos en los cuales viajó

$(\forall p \in PASAJEROS) (\forall i \in ITINERARIOS)$
 $(\langle p, i \rangle \in eval \rightarrow \langle p, i \rangle \in p_i)$

La hora de llegada de un itinerario debe de ser mayor a la hora de salida

$(\forall i \in ITINERARIOS) (hora_llegada(i) > hora_salida(i))$

El personal de tierra esta asociado a itinerarios vinculados a puertas de embarque del mismo aeropuerto donde opera el itinerario.

$(\forall i \in ITINERARIOS) (\forall a \in AEROPUERTOS) (\forall pt \in PERSONALTIERRA)$
 $(\forall pe \in PUERTASEMBARQUE)$

$(\langle i, a \rangle \in opera \wedge \langle a, pe \rangle \in en \wedge \langle pt, i \rangle \in pt_i) \rightarrow (pe \in \langle pe_i \rangle \wedge \langle i, pe \rangle \in \langle pe_i \rangle)$

Ejercicio 2 (26 puntos)

Una emisora de radio maneja información acerca de su programación, locutores y anunciantes. El esquema de la base de datos es el siguiente:

PROGRAMAS (idPrograma, nombrePrograma, duracion, tipo)

ANUNCIANTES (idAnunciante, nombreAnunciante, sector)

ANUNCIOS (idAnuncio, fecha, horario, idAnunciante, duracion)

LOCUTORES (idLocutor, nombreLocutor, especialidad)

PROGRAMACION (idPrograma, idLocutor, fecha, horario)

En este esquema no existen tablas vacías. Además, sabemos que los anuncios correspondientes a un programa son los que se emiten en la misma fecha y horario que el programa.

Se cumplen las siguientes dependencias de inclusión:

- $\Pi_{\text{idPrograma}}(\text{PROGRAMACION}) \subseteq \Pi_{\text{idPrograma}}(\text{PROGRAMAS})$
- $\Pi_{\text{idLocutor}}(\text{PROGRAMACION}) \subseteq \Pi_{\text{idLocutor}}(\text{LOCUTORES})$
- $\Pi_{\text{idAnunciante}}(\text{ANUNCIOS}) \subseteq \Pi_{\text{idAnunciante}}(\text{ANUNCIANTES})$

Resolver el álgebra relacional las siguientes consultas

1. Devolver todos los datos de los anuncios que se emitieron en horarios donde no hubo ningún programa de tipo 'Musical'.

Solución:

$PM = \Pi_{\text{idPrograma}}(\sigma_{\text{tipo} = \text{"Musical"}}(\text{PROGRAMAS}))$

$HNM = \Pi_{\text{horario}}(\text{ANUNCIOS}) - \Pi_{\text{horario}}(\text{ANUNCIOS} * \text{PROGRAMACION} * PM)$

$RES = \text{ANUNCIOS} * HNM$

2. Devolver el nombre de los locutores que han trabajado en todos los programas de más de 1 hora de duración.

Solución:

$P1H = \Pi_{\text{idPrograma}}(\sigma_{\text{duración} > 1}(\text{PROGRAMAS}))$

$L = \Pi_{\text{idLocutor}, \text{idPrograma}}(\text{PROGRAMACION}) \% P1H$

$RES = \Pi_{\text{nombreLocutor}}(\text{LOCUTORES} * L)$

Resolver el cálculo relacional la siguiente consulta

3. Devolver los nombres de programas del tipo 'Informativo' en los que **solo** han anunciado anunciantes del sector 'Tecnología'

Solución:

$$\{ \langle p.\text{nombrePrograma} \rangle / \text{PROGRAMAS}(p) \wedge p.\text{tipo} = \text{"Informativo"} \wedge \\ \neg(\exists pg) (\text{PROGRAMACIÓN}(pg) \wedge p.\text{idPrograma} = pg.\text{idPrograma} \wedge \\ (\exists a)(\exists an) (\text{ANUNCIOS}(a) \wedge \text{ANUNCIANTES}(an) \wedge an.\text{sector} \neq \text{"Tecnología"} \wedge \\ a.\text{idAnunciante} = an.\text{idAnunciante} \wedge a.\text{fecha} = pg.\text{fecha} \wedge a.\text{horario} = pg.\text{horario})) \wedge \\ (\exists pg_2) (\text{PROGRAMACIÓN}(pg_2) \wedge p.\text{idPrograma} = pg_2.\text{idPrograma} \wedge \\ (\exists a_2)(\exists an_2) (\text{ANUNCIOS}(a_2) \wedge \text{ANUNCIANTES}(an_2) \wedge an_2.\text{sector} = \text{"Tecnología"} \wedge \\ a_2.\text{idAnunciante} = an_2.\text{idAnunciante} \wedge a_2.\text{fecha} = pg_2.\text{fecha} \wedge a_2.\text{horario} = pg_2.\text{horario})) \}$$

Resolver en SQL la siguiente consulta

4. Devolver el nombre del anunciante, la cantidad de anuncios que ha emitido y el promedio de la duración de sus anuncios, para aquellos anunciantes que han emitido la mayor cantidad de anuncios.

Solución:

```
SELECT nombreAnunciante, COUNT(*), AVG(duración)
FROM ANUNCIANTES NATURAL JOIN ANUNCIOS
GROUP BY idAnunciante
HAVING COUNT(*) >= ALL (
    SELECT COUNT(*)
    FROM ANUNCIANTES NATURAL JOIN ANUNCIOS
    GROUP BY idAnunciante
)
```

Ejercicio 3 (27 puntos)

Se considera una relación R con un conjunto de dependencias funcionales F dado por:

$R(A, B, C, D, E, G, H)$ y $F = \{ AC \rightarrow B, BE \rightarrow H, AG \rightarrow D, BH \rightarrow E, AD \rightarrow G \}$

1. **Justificando su respuesta, indique cuáles de las siguientes dependencias están en F^+ y cuáles no.**

- a) $AE \rightarrow B$
- b) $ACH \rightarrow E$
- c) $AB \rightarrow C$

Para verificar si una dependencia se cumple en F^+ haremos la clausura de su lado izquierdo, según F. Si dentro de la clausura encontramos los atributos del lado derecho podemos afirmar que la dependencia se cumple en F^+ .

$(AE)^+ = \{A, E\}$. Como $B \notin (AE)^+$ podemos afirmar que $AE \rightarrow B$ no se cumple en F^+

$(ACH)^+ = \{A, C, H, B, E\}$. Como $E \in (ACH)^+$ podemos afirmar que $ACH \rightarrow E$ se cumple en F^+

$(AB)^+ = \{A, B\}$. Como $C \notin (AB)^+$ podemos afirmar que $AB \rightarrow C$ no se cumple en F^+

2. **Hallar todas las claves de R. Justifique.**

Para hallar todas las claves de R comenzamos determinando el conjunto de atributos que está en toda clave (los que sólo aparecen a la izquierda en Dfs o no aparecen en ninguna) y el de los que no están en ninguna clave (sólo aparecen a la derecha)

En toda clave: $\{A, C\}$

En ninguna: $\{\}$

Probamos si alcanza con (AC) para determinar todos los atributos de R, y para eso calculamos $(AC)^+$

$(AC)^+ = \{A, C, B\}$. Como faltan atributos de R sabemos que AC no es clave, pero tiene que estar en toda clave.

Por lo tanto la/s claves contienen a AC y tienen más atributos.

Podemos observar que la única df con E a la derecha es $BH \rightarrow E$, por lo tanto para determinar a E o lo incluimos directamente o agregamos a H a la clave. Un razonamiento similar se puede hacer con respecto a H debido a $BE \rightarrow H$. Por lo tanto las claves candidatas deberán contener a E o a H. Algo similar ocurre con D y G debido a que solo aparecen a la derecha en $AG \rightarrow D, AD \rightarrow G$

En resumen, las claves candidatas son: ACEG, ACED, ACHG y ACHD

$(ACEG)^+ = \{A, C, B, E, G, D, H\}$

$(ACED)^+ = \{A, C, B, E, D, H, G\}$

$(ACHG)^+ = \{A, C, B, H, E, G, D\}$

$(ACHD)^+ = \{A, C, B, H, E, D, G\}$

Estas son las cuatro únicas claves de R xq se puede observar que si existiera alguna clave de 5 atributos debería contener a AC y no contener a alguna de las claves ya halladas lo cual es

imposible porque R tiene 7 atributos Veamos por ejemplo para ACEG

$\{AC\} \cup R - \{ACEG\} = \{A,C,B,D,H\}$ contiene a ACHD

3. Determinar si F es un conjunto minimal. Si no lo es, hallar un cubrimiento minimal de F. Si lo es, hallar un conjunto K equivalente a F, que no sea minimal. Justifique sus respuestas.
4. Aplicando el algoritmo visto en el curso, llevar R a 3NF con join sin pérdida y preservación de dependencias. Explique cada uno de los pasos aplicados.
5. Dada la descomposición de R: $\rho = \{ R1(A,B,C,D), R2(A,D,E,G,H) \}$, realizar lo siguiente:
 - a) Para cada R_i , indicar en qué forma normal se encuentra. Justifique.
 - b) Decidir si es posible que se cumpla la dependencia multivaluada $D \twoheadrightarrow AC$ en $R1$. Justifique.

3. Determinar si F es un conjunto minimal. Si no lo es, hallar un cubrimiento minimal de F. Si lo es, hallar un conjunto K equivalente a F, que no sea minimal. Justifique sus respuestas.

Verificamos que las Dfs de $F = \{ AC \rightarrow B, BE \rightarrow H, AG \rightarrow D, BH \rightarrow E, AD \rightarrow G \}$ cumplan con los criterios de minimalidad:

1 - Toda df en F tiene un solo atributo a la derecha

se cumple

2 - No podemos reemplazar ninguna $df X \rightarrow A \in F$ por una $df Y \rightarrow A$, donde $Y \subset X$, y seguir teniendo un conjunto de dfs equivalente a F

Observamos que los atributos que aparecen a la derecha sólo aparecen a la derecha en una df, por lo tanto no será posible encontrar atributos redundantes en las Dfs de F (si elimino a cualquiera de ellos no será posible alcanzar el lado derecho). Por lo tanto, cumple que no hay atributos redundantes.

3- No podemos quitar ninguna df de F y seguir teniendo un conjunto de dfs equivalente a F

Por el mismo argumento usado en el punto 2 podemos observar que no hay Dfs redundantes.

Como se cumplen las tres podemos decir que F es minimal.

Ahora necesitamos hallar un K equivalente a F que no sea minimal, o sea que no cumpla alguno de los puntos anteriores. Agregaremos alguna dependencia redundante, por ejemplo alguna de las que se desprenden de considerar $(ACE)^+_{en F} = \{A, C, B, E, H\}$, como $ACE \rightarrow H$

En resumen, $K = F \cup \{ ACE \rightarrow H \}$ no es minimal y es equivalente a F porque la dependencia agregada se obtuvo mirando la clausura en F de un conjunto de atributos.

4. Aplicando el algoritmo visto en el curso, llevar R a 3NF con join sin pérdida y preservación de dependencias. Explique cada uno de los pasos aplicados.

El algoritmo indica los siguientes pasos:

i. encontrar un cubrimiento minimal G para F;

Ya probamos en el paso anterior que F es minimal, por lo tanto haremos $G=F$.

ii. para cada miembro izq X de una df que aparezca en G

crear un er $\{X \cup A_1 \cup A_2 \dots \cup A_m\}$ en D, donde $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_m$ sean todas las dfs en G con X como miembro izq;

Como en F no se repiten los lados izquierdos generamos cinco esquemas relacionales, uno por DF

R1(ACB)

R2(BEH)

R3(AGD)

R4(BHE)

R5(ADG)

iii. si ninguno de los R_i contiene una clave de R , crear un R_i adicional que contenga atributos que formen una clave de R ;

Agregamos R6(ACEG)

iv. eliminar R_i redundantes, es decir que estén contenidos en otro ya generado.

R2 y R4 son iguales, R3 y R5 son iguales, así que $\rho_1 = \{ R1(ACB), R2(BEH), R3(AGD), R4(ACEG) \}$ es una descomposición de R en 3NF con preservación de dependencias y JSP.

5- a) Para indicar en qué forma normal se encuentra cada R_i comenzamos por proyectar las Dfs de $F = \{ AC \rightarrow B, BE \rightarrow H, AG \rightarrow D, BH \rightarrow E, AD \rightarrow G \}$

R1(A,B,C,D)

$\Pi_{R1}(F) = \{ AC \rightarrow B \}$

CLAVES de R1: Como D no está en ninguna D_f , entonces está en toda clave. DAC es la única clave.

Como AC no es superclave, R1 no está en BCNF.

Para que R1 esté en 3NF debería cumplirse que B sea primo (miembro de alguna clave). Como la única clave es DAC esto no se cumple y R1 no está en 3NF.

La dependencia $AC \rightarrow B$ es una dependencia parcial de la clave DAC, por lo que R1 tampoco está en 2NF. Finalmente, R1 está en 1NF

R2(A,D,E,G,H)

$\Pi_{R2}(F) = \{ AG \rightarrow D, AD \rightarrow G \}$

CLAVES: E y H están en toda clave por que no están en ninguna D_f . (EHAD) y (EHAG) son las únicas claves.

R2 no está en BCNF porque AG y AD no son superclaves, pero si está en 3NF porque D y G son primos.

5 b) Decidir si es posible que se cumpla la dependencia multivaluada $D \twoheadrightarrow AC$ en R1. Justifique.

Para que sea posible que se cumpla esta DMV, y de acuerdo con la definición[*] tendría que ser posible tener instancias como la siguiente:

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d1
a1	b2	c1	d1
a2	b1	c2	d1

Pero por la parte anterior sabemos que $\prod_{R_1}(F) = \{AC \rightarrow B\}$, por lo tanto no es posible tener instancias que tengan igual valor en AC y diferente en B (por definición de DF). Por lo tanto, no es posible que se cumpla la DMV indicada en R1

[*] Una dmv $X \twoheadrightarrow Y$ especificada sobre el er R (X e Y subconj de R) especifica la siguiente restricción sobre cualquier relación r de R:

Si existen 2 tuplas t1 y t2 en r tales que $t1(X) = t2(X)$, entonces deberán existir también 2 tuplas t3 y t4 en r con las siguientes propiedades:

- $t3(X) = t4(X) = t1(X) = t2(X)$
- $t3(Y) = t1(Y)$ y $t4(Y) = t2(Y)$
- $t3[R - (XY)] = t2[R - (XY)]$ y $t4[R - (XY)] = t1[R - (XY)]$

Ejercicio 4 (20 puntos)

Dadas las siguientes transacciones:

T1: w1(Z) r1(Y) w1(X) w1(Y) c1

T2: r2(Y) r2(X) w2(Y) w2(X) w2(Z) c2

Parte A

Para cada una de las siguientes afirmaciones, indique si son verdaderas o falsas. **Justifique.**

- a) La siguiente historia no evita abortos en cascada
w1(Z) r2(Y) r2(X) r1(Y) w1(X) w1(Y) c1 w2(Y) w2(X) w2(Z) c2

Falso

Una historia evita abortos en cascada si ninguna transacción lee de transacciones no confirmadas

w1(Z) r2(Y) r2(X) r1(Y) w1(X) w1(Y) c1 w2(Y) w2(X) w2(Z) c2

En esta historia, T₂ no lee de T₁. Por lo tanto esta historia evita abortos en cascada.

- b) La siguiente historia es estricta
w1(Z) r2(Y) r1(Y) w1(X) w1(Y) c1 r2(X) w2(Y) w2(X) w2(Z) c2

Verdadero

Una historia es estricta si ninguna transacción lee o escribe hasta que todas las transacciones que escribieron ese ítem fueron confirmadas.

En esta historia ni T₁ ni T₂ leen o escriben ítems previamente escritos y no confirmados por la otra transacción.

Los siguientes pares de operaciones:

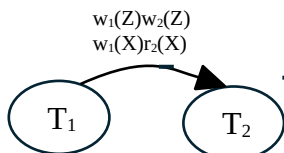
w₁(Z), w₂(Z)
w₁(X), w₂(X)
w₁(X), r₂(X)

No violan la condición, pues entre ellas esta el commit de T₁

- c) La siguiente historia es serializable
w1(Z) r1(Y) w1(X) w1(Y) r2(Y) r2(X) c1 w2(Y) w2(X) w2(Z) c2

Verdadero

Vamos a construir el grafo de seriabilidad:



Dado que el grafo no presenta ningún ciclo, podemos afirmar que la historia es serializable.

Parte B

a) Escriba T1 y T2, incorporando bloqueos de escritura y lectura, de forma de que una de las transacciones siga 2PL básico.

En 2PL básico sólo se exige que estén presentes las fases de crecimiento y contracción

T1: $w_1(Z)$ $w_1(Z)$ $w_1(Y)$ $r_1(Y)$ $w_1(X)$ $w_1(X)$ $w_1(Y)$ $u_1(Z)$ $u_1(Y)$ $u_1(X)$ c_1

T2:

b) Escriba una historia válida y entrelazada de las T1 y T2 de la parte a).

H: $w_1(Z)$ $w_1(Z)$ $w_1(Y)$ $r_1(Y)$ $w_1(X)$ $w_1(X)$ $u_1(X)$ $w_2(X)$ $r_2(X)$ $w_1(Y)$ $u_1(Z)$ $u_1(Y)$ $w_2(Y)$ $w_2(X)$ $w_2(Z)$
 $w_2(Z)$ $u_2(Y)$ $u_2(X)$ $u_2(Z)$ c_1 c_2