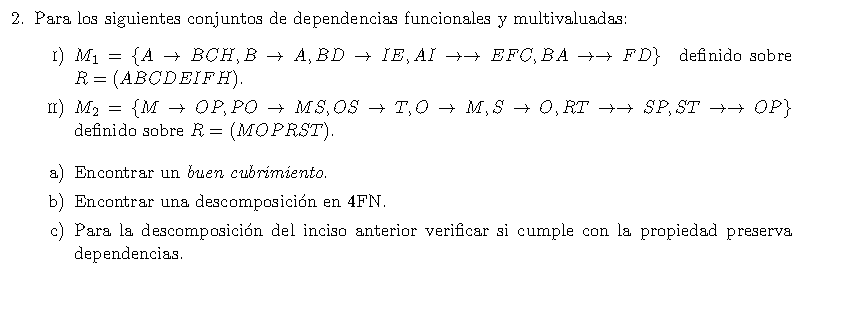
**Multivaluadas**



2)i) M1 = {A🡪BCH, B🡪A, BD🡪IE, AI🡪🡪EFC, BA🡪🡪FD} R = (ABCDEIFH)

**1. Abrir df’s a derecha**

M2 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪EFC, BA🡪🡪FD}

**2.Abrir dependencias multivaluadas a derecha**

**AI🡪🡪EFC**

Vamos a hacer la base de dependencias para la dm (dependencia multivaluada) AI🡪🡪EFC

(agregamos la R y le sacamos AI)

Objetivo: Separar las letras de la derecha de la dm (en este caso EFC)

Este camino es una pinga y no me lleva a ningún lado

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A🡪C |  |  |  |  |  |  |  |
| B**AI**(M2)=(BCDEFH) | = | (C,BDEFH) | = |  |  |  |  |  |

Pruebo otro camino

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | AI🡪🡪EFC |  | A🡪C | \*\*\* | BD🡪E |  |  |  |
| B**AI**(M2)=(BCDEFH) | = | (EFC,BDH) | = | (C,**EF**,BDH) | = | (C,E,F,BDH) |  |  |

\*\*\*Ahora solo me interesa separar las letras del bloque EF el resto me las paso por los huevos.

Luego M3 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪E, AI🡪🡪F, AI🡪🡪C, BA🡪🡪FD}

M3 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪E, AI🡪🡪F, AI🡪🡪C, BA🡪🡪FD}

Abrimos BA🡪🡪FD

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | BA🡪🡪FD |  | AI🡪🡪F |  |  |  |  |  |
| B**BA**(M3)=(CDEIFH) | = | (FD,CEIFH) | = | (F,D,CEIFH) |  |  |  |  |

Luego M4 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪E, AI🡪🡪F, AI🡪🡪C, BA🡪🡪F, BA🡪🡪D}

**3.Eliminar dependencias redundantes (primero chequeamos si hay dfs que tengan el mismo lado derecho, formalmente: estamos buscando dfs en M4 que verifiquen la 2da condición de Beeri)**

Por su condición de df, se tienen que cumplir las dos condiciones del Teorema de Beeri: B debe ser un atributo simple en BA(G 4 \ {A🡪B}) y existir una alguna (otra) df que tenga a B en el lado derecho.

–Como no se cumple la segunda condición, la df A🡪B no es redundante y no hace falta calcular las bases de dependencia.

–Lo mismo sucede con las df’s A🡪C, A🡪H, B🡪A, BD🡪I y BD🡪E.

En este caso no hay dependencias redundantes

**Chequeamos si hay dm redundantes (el objetivo es separar todas las letras del esquema)**

Ahora no podemos utilizar la propia dm

¿AI🡪🡪E es redundante? Utilizar todas las dependencias menos la que queremos ver si es redundante

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪B |  | A🡪C |  | A🡪H |  |  |
| (AI)**M4\{AI🡪🡪E}** | B**AI**(M4) =(BCDEFH) | = | (B,CDEFH) | = | (B,C,DEFH) | = |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | AI🡪🡪F |  | BA🡪🡪D |  |  |  |  |  |
| (B,C,H,DEF) | = | (B,C,H,F,DE) | = | (B,C,H,F,D,E) |  |  |  |  |

Luego AI🡪🡪E es redundante.

Luego M5 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪F, AI🡪🡪C, BA🡪🡪F, BA🡪🡪D}

¿AI🡪🡪F es redundante?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪B |  | A🡪C |  | A🡪H |  |  |
| (AI)**M5\{AI🡪🡪F}** | B**AI**(M5) =(BCDEFH) | = | (B,CDEFH) | = | (B,C,DEFH) | = |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | BA🡪🡪F |  | BA🡪🡪D |  |  |  |  |  |
| (B,C,H,DEF) | = | (B,C,H,F,DE) | = | (B,C,H,F,D,E) |  |  |  |  |

Luego AI🡪🡪F es redundante.

M6 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, AI🡪🡪C, BA🡪🡪F, BA🡪🡪D}

¿AI🡪🡪C es redundante?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪B |  | A🡪C |  | A🡪H |  |  |
| (AI)**M6\{AI🡪🡪C}** | B**AI**(M6) =(BCDEFH) | = | (B,CDEFH) | = | (B,C,DEFH) | = |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | BA🡪🡪F |  | BA🡪🡪D |  |  |  |  |  |
| (B,C,H,DEF) | = | (B,C,H,F,DE) | = | (B,C,H,F,D,E) |  |  |  |  |

Luego AI🡪🡪C es redundante

M7 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, BA🡪🡪F, BA🡪🡪D}

BA🡪🡪F es redundante?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪C |  | A🡪H |  | BA🡪🡪D |  |  |
| (BA)**M7\{BA🡪🡪F}** | B**BA**(M7) =(CDEIFH) | = | (C,DEIFH) | = | (C,H,DEIF) | = |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | BD🡪E |  | BD🡪I |  |  |  |  |  |
| (C,H,D,EIF) | = | (C,H,D,E,IF) | = | (C,H,D,E,I,F) |  |  |  |  |

Luego BA🡪🡪F es redundante

Luego M8 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, BA🡪🡪D}

BA🡪🡪D es redundante?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪C |  | A🡪H |  |  |  |  |
| (BA)**M8\{BA🡪🡪D}** | B**BA**(M8) =(CDEIFH) | = | (C,DEIFH) | = | (C,H,DEIF) | = |  |  |

Este camino no me llevo a ningún lado, tampoco hay más caminos disponibles.

BA🡪🡪D no es redundante

**4.Eliminar atributos extraños a izquierda (agarramos las dependencias que tengan más atributos)**

**Objetivo: Separar el lado derecho de la dependencia.**

M8 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, BA🡪🡪D}

BD🡪I

¿B es extraño a izquierda?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| B: | B**D**(M8) =(ABCEIFH) | = |  | = |  | = |  |  |

No se puede hacer nada en este caso por lo tanto B no es extraño a izquierda

M8 = {A🡪B, A🡪C, A🡪H, B🡪A, BD🡪I, BD🡪E, BA🡪🡪D}

¿D es extraño a izquierda?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | B🡪A |  | A🡪C |  | A🡪H |  |  |
| D: | B**B**(M8) =(ACDEIFH) | = | (A,CDEIFH) | = | (A,C,DEIFH) | = | (A,C,H,DEIF) |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BA🡪🡪D |  | BD🡪E |  | BD🡪I |  |  |  |  |
| = | (A,C,H,D,EIF) | = | (A,C,H,D,E,IF) | = | (A,C,H,D,E,I,F) |  |  |  |

Por lo tanto, D es extraño a izquierda, entonces lo elimino y queda B🡪I.

Observación: Podría haberlo hecho en menos pasos siguiendo B🡪A; BA🡪🡪D; BD🡪I.

M9 = {A🡪B, A🡪C, A🡪H, B🡪A, B🡪I, BD🡪E, BA🡪🡪D}

BD🡪E

¿B es extraño a izquierda?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| B: | B**D**(M9) =(ABCEIFH) | = |  | = |  | = |  |  |

No se puede calcular la base de dependencia de BD🡪E para el literal B, por lo tanto, B no es extraño a izquierda.

¿D es extraño a izquierda?

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | B🡪A |  | BA🡪🡪D |  | BD🡪E |  |
| D: | B**B**(M9) =(ACDEIFH) | = | (A,CDEIFH) | = | (A,D,CEIFH) | = | (A,D,E,CIFH) |

Como pudimos separar la “E” entonces D es extraño a izquierda.

M10 = {A🡪B, A🡪C, A🡪H, B🡪A, B🡪I, B🡪E, BA🡪🡪D}

BA🡪🡪D

¿B es extraño a izquierda?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A🡪B |  | BA🡪🡪D |  |  |  |  |
| B: | B**A**(M10) =(BCDEIFH) | = | (B,CDEIFH) | = | (B,D,CEIFH) | = |  |  |

Como pude separar la D, B es extraño a izquierda.

M11 = {A🡪B, A🡪C, A🡪H, B🡪A, B🡪I, B🡪E, A🡪🡪D}

5. Fusionar las df’s con los mismos lados izquierdos

M12 = {A🡪BCH, B🡪AIE, A🡪🡪D}

Luego M12 es un buen cubrimiento para M1.

**Cálculo de llaves candidatas** para M12= {A🡪BCH, B🡪AIE, A🡪🡪D} R =(ABCDEIFH) (la multivaluada no se tiene en cuenta para este primer paso)

|  |  |  |
| --- | --- | --- |
| Siempre | Tal vez | Nunca |
| FD | AB |  |

FDM12\*=FD **⊉ R**

Pruebo con 3 atributos:

FDAM12\*= FDABCHIE ⊇ R por lo tanto es llave

FDBM12\*= FDBAIECH ⊇ R por lo tanto es llave

Luego calculo la base de dependencia de aquellos esquemas que no son llave.

M12= {A🡪BCH, B🡪AIE, A🡪🡪D} R =(ABCDEIFH)

Al conjunto R le sacamos los literales que están en “Siempre”. Objetivo separar todas las letras

|  |  |  |
| --- | --- | --- |
|  |  |  |
| FD | B**FD**(M12) =(ABCEIH) |  |

Como no pudimos separar todas las letras FD no es llave.

Entonces las llaves candidatas para M12 son FDA y FDB.