

Representación de Pfaltz

Concepto de grafo bipartito

Es un grafo en el cual el conjunto de puntos se divide en dos subconjuntos disjuntos, de tal manera que no exista un arco entre dos nodos que pertenezcan a un mismo conjunto.

Dado $G(P,E)$, G es bipartito si:

- $P = Q \cup R$
- $Q \cap R = \emptyset$
- $\forall (x,y) \in E: (x \in Q \wedge y \in R) \vee (x \in R \wedge y \in Q)$

Forma algebraica de la representación de Pfaltz

Pfaltz plantea una forma de representación de grafos irrestrictos de manera que se puedan representar los grafos completos con sus asignaciones a puntos y a arcos.

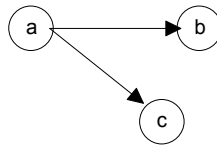
Pfaltz pasa de $G(P,E)$ a un $G'(P',E')$, haciendo que cada arco (elemento de E) pase a ser también un nodo en P' . El conjunto E' contendrá dos relaciones por cada una en el conjunto E , original: la primer componente del arco relacionada con el arco, y el arco relacionada con la segunda componente. Veamos este concepto aplicado al siguiente ejemplo:

Sea:

$G(P,E)$ una $E_d(P,E, f_1, f_2, \dots, f_n, g_1, g_2, \dots, g_n)$.

$P = \{a, b, c\}$

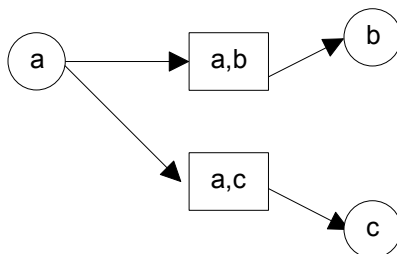
$E = \{(a,b), (c,d)\}$



Entonces su representación algebraica de Pfaltz $G'(P',E')$

$P' = P \cup E = \{a, b, c, (a,b), (a,c)\}$

$E' = \{(a, (a,b)), ((a,b), b), (a, (a,c)), ((a,c), c)\}$



Observar que :

$$(x,y) \in E \Leftrightarrow (x,(x,y)) \in E' \wedge ((x,y),y) \in E'$$

Forma computacional de la representación de Pfaltz

Hasta el momento conocíamos dos formas de representar un grafo irrestricto: las matrices de incidencia y adyacencia. Las representaciones en forma de matriz para un grafo no resultan del todo adecuadas cuando se aplican a problemas reales, dado que su uso requiere el conocimiento a priori de la cantidad de nodos y/o de relaciones.

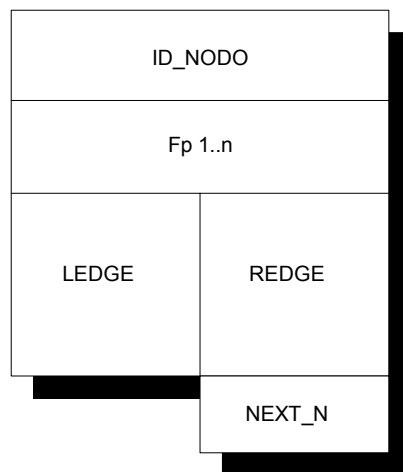
Supóngase que el problema a resolver mediante la computadora implica la creación o destrucción de nodos o arcos dinámicamente a medida que se avanza en la solución del problema, lo que implica la actualización dinámica de la estructura de datos computacional durante el programa. Si esta fuese una matriz, esta debería ser creada nuevamente para cada alta o baja de un nodo; para el caso de un alta o baja de un arco pasaría lo mismo si se trabajase con una matriz de incidencia.

Para que esto no ocurra, Pfaltz trabaja con una representación dinámica de nodos y arcos enlazados mediante punteros, ocupando espacio en memoria a medida que realmente se necesita (cuando se crea un nuevo nodo o un nuevo arco), y liberándolo cuando se efectúa una baja.

Para un problema medianamente complejo donde se pueden llegar a tener cientos de nodos, la representación de Pfaltz es más eficiente que el uso de matrices. Una matriz sería extremadamente grande teniendo en cuenta que siempre se reserva espacio para las relaciones, independientemente de la cantidad de arcos realmente existentes. Planteemos por ejemplo el tiempo que se tardaría en realizar la baja de un nodo en el grafo.

Las estructuras de datos que utiliza Pfaltz son básicamente dos listas, una de nodos, y otra de arcos, con el diseño de celda que se muestra a continuación:

Estructura de Celda de tipo "NODO"



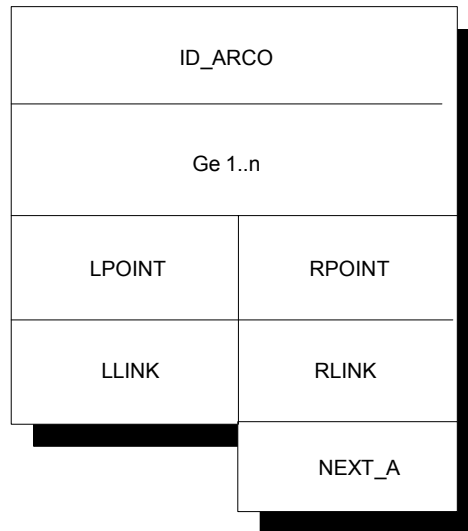
Detalle de Campos

- **ID_NODO** Identificador único del nodo.
- **Fp 1..n** Funciones de asignación del nodo.
- **LEDGE** Puntero con la dirección de la primer celda de tipo "ARCO" de una lista de arcos que llegan al nodo.
- **REDGE** Puntero con la dirección de la primer celda de tipo "ARCO" de una

lista de arcos que parten del nodo.

- **NEXT_N** *Puntero que apunta a la próxima celda de tipo “NODO” de la lista de nodos ingresada.*

Estructura de Celda de tipo “ARCO”

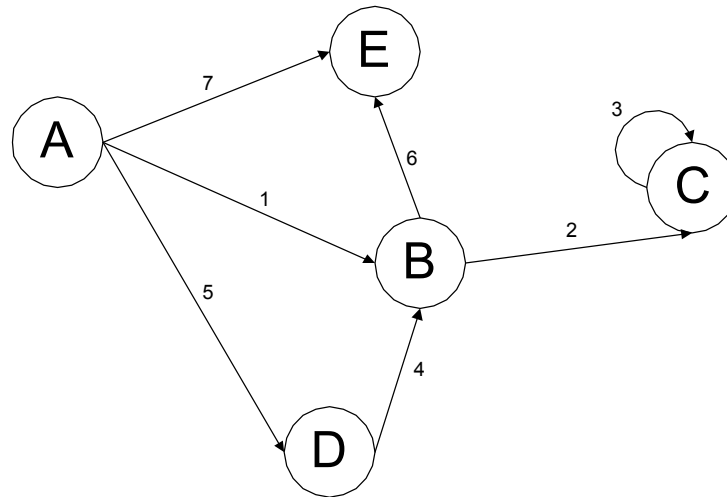


Detalle de Campos

- **ID_ARCO** *Identificador único del arco.*
- **Ge 1..n** *Funciones de asignación del arco.*
- **LPOINT** *Puntero con la dirección de la celda de tipo “NODO” de la cuál parte el arco.*
- **RPOINT** *Puntero con la dirección de la celda de tipo “NODO” a la cuál llega el arco.*
- **LLINK** *Puntero con la dirección de la próxima celda de tipo “ARCO” de una lista de arcos con igual RPOINT (o sea que llega al mismo nodo).*
- **RLINK** *Puntero con la dirección de la próxima celda de tipo “ARCO” de una lista de arcos con igual LPOINT (o sea que parte del mismo nodo).*
- **NEXT_A** *Puntero que apunta a la próxima celda de tipo “ARCO” de la lista de arcos ingresada.*

Caso Práctico

Dado el siguiente grafo $G(P,E)$



Se procederá a implementarlo en un modelo computacional basado en la representación de Pfaltz.

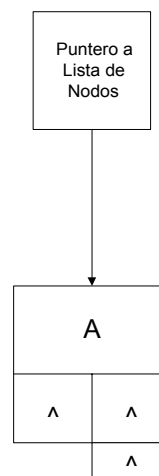
La implementación del mismo se realizará en dos etapas, en la primera se ingresarán todos los nodos y en la segunda etapa las relaciones.

Ingreso de Nodos

Los nodos se ingresarán en el siguiente orden A,B,C,D,E.

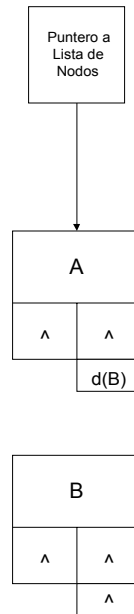
Se ingresa el nodo A en la estructura.

- Se actualiza un puntero externo a la estructura con la dirección del nodo A.



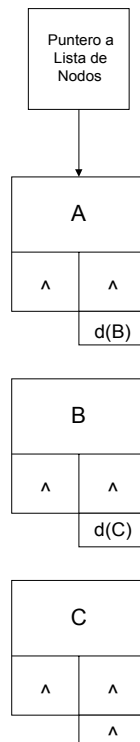
Ingreso del Nodo B en la estructura.

- Se actualiza el puntero next_n de la última celda de tipo nodo ingresada (A) con la dirección de la celda del nuevo nodo B.



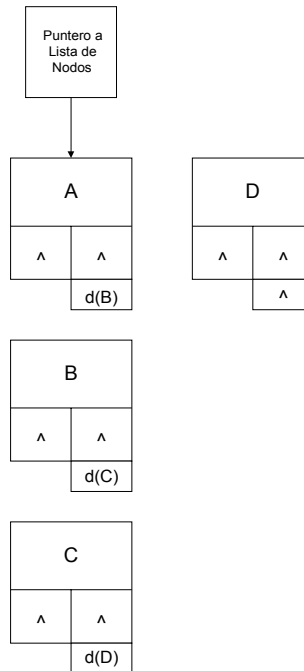
Ingreso del Nodo C a la estructura

- Se actualiza el puntero next_n de la última celda de tipo nodo ingresada (B) con la dirección de la celda del nuevo nodo C.



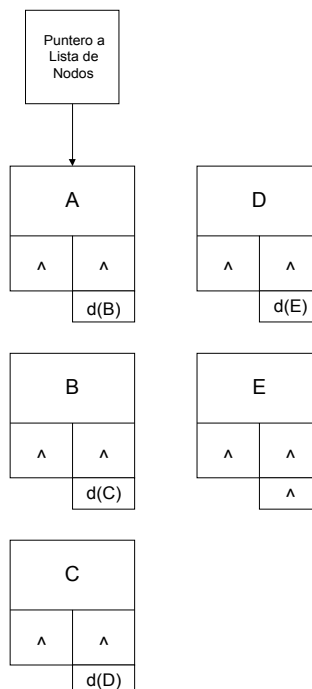
Ingreso del Nodo D a la estructura

- Se actualiza el puntero next_n de la última celda de tipo nodo ingresada (C) con la dirección de la celda del nuevo nodo D.



Ingreso del Nodo E a la estructura

- Se actualiza el puntero next_n de la última celda de tipo nodo ingresada (D) con la dirección de la celda del nuevo nodo E.



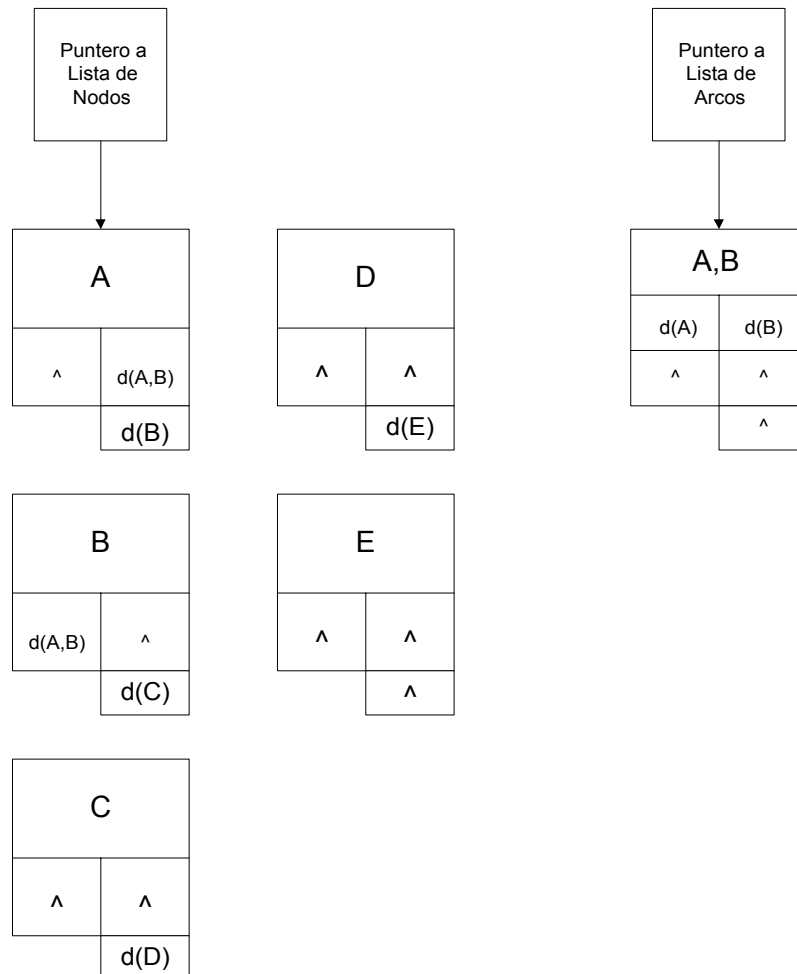
Ingreso de Relaciones

El orden de ingreso de las relaciones será el siguiente:

(A,B);(B,C);(C,C);(D,B);(A,D);(B,E);(A,E)

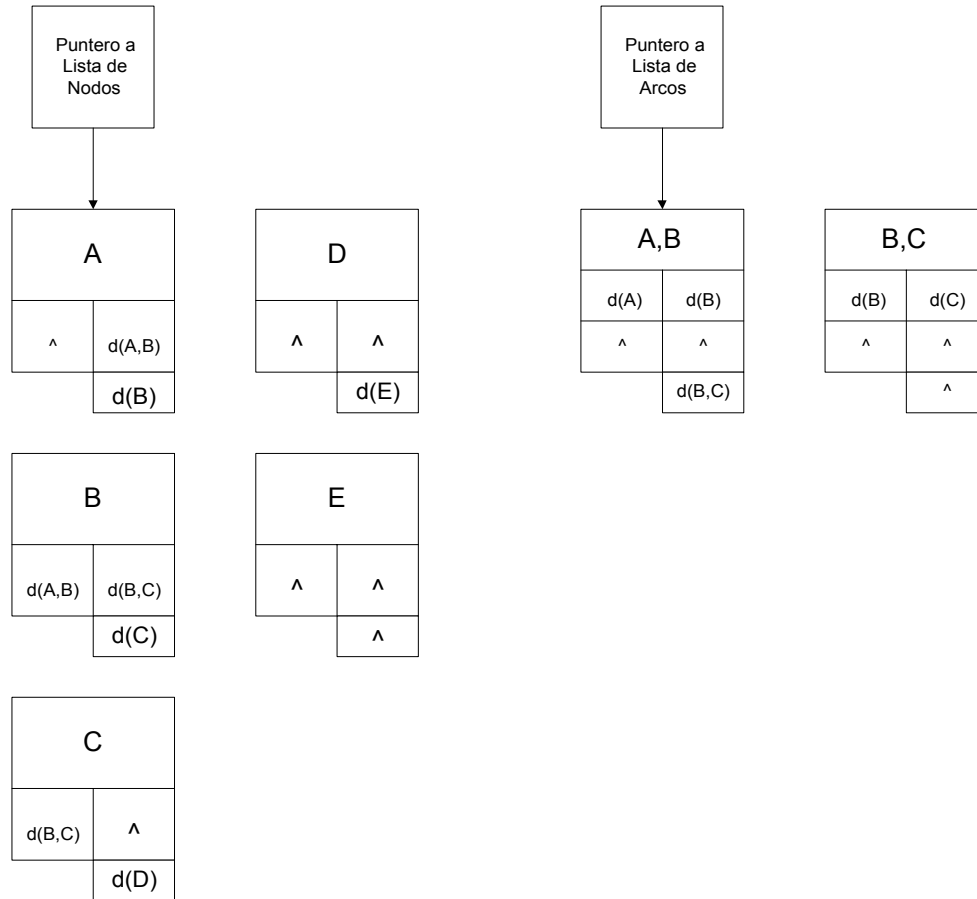
Ingreso de la relación (A,B)

- Se actualiza un puntero externo a la estructura con la dirección de (A,B).
- En la celda tipo arco "A,B" se actualizan lpoint = d(A) y rpoint = d(B)
- En la celda tipo nodo "A" se actualiza el redge = d(A,B) por ser el primer arco que parte de "A".
- En la celda tipo nodo "B" se actualiza el ledge = d(A,B) por ser el primer arco que llega a "B".



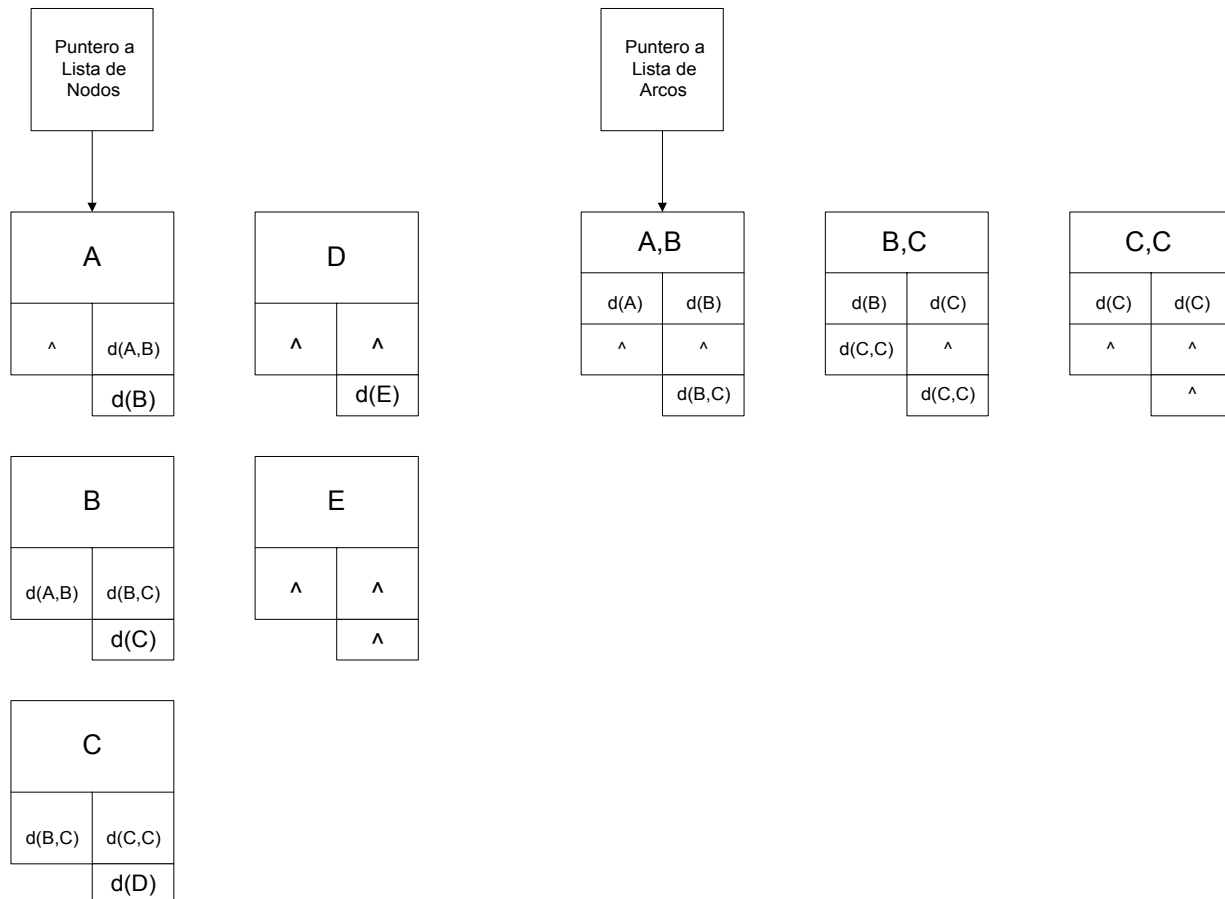
Ingreso de la relación (B,C)

- En la celda de tipo arco "A,B" se actualiza el next_a = d(B,C) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco "B,C" se actualizan lpoint = d(B) y rpoint = d(C)
- En la celda tipo nodo "B" se actualiza el redge = d(B,C) por ser el primer arco que parte de "B".
- En la celda tipo nodo "C" se actualiza el ledge = d(B,C) por ser el primer arco que llega a "C".



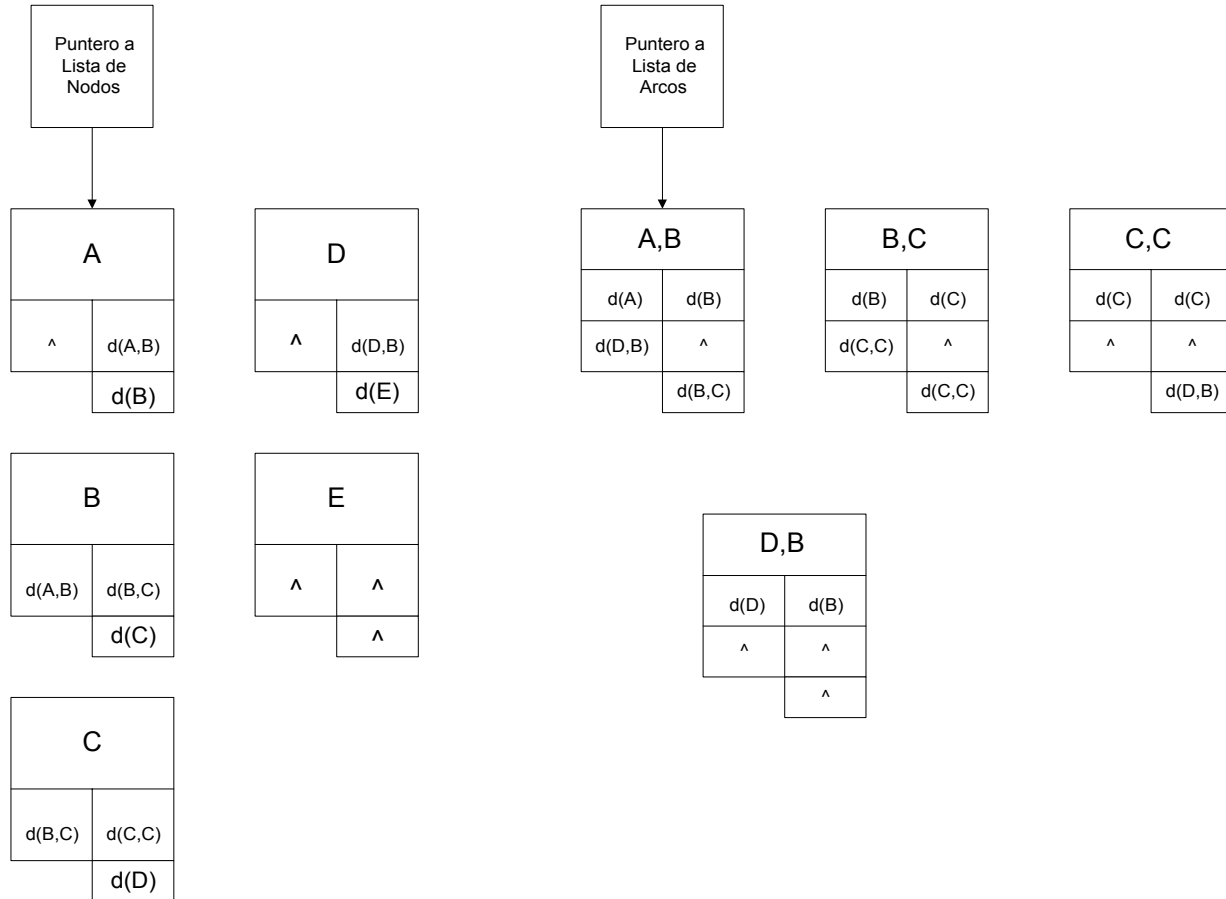
Ingreso de la relación (C,C)

- En la celda de tipo arco "B,C" se actualiza el next_a = d(C,C) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco "C,C" se actualizan lpoint = d(C) y rpoint = d(C)
- En la celda tipo nodo "C" se actualiza el redge = d(C,C) por ser el primer arco que parte de "C".
- En la celda tipo arco "B,C" se actualiza el llink = d(C,C) por ser este el próximo arco que llega a "C", o sea que tiene igual rpoint que "B,C".



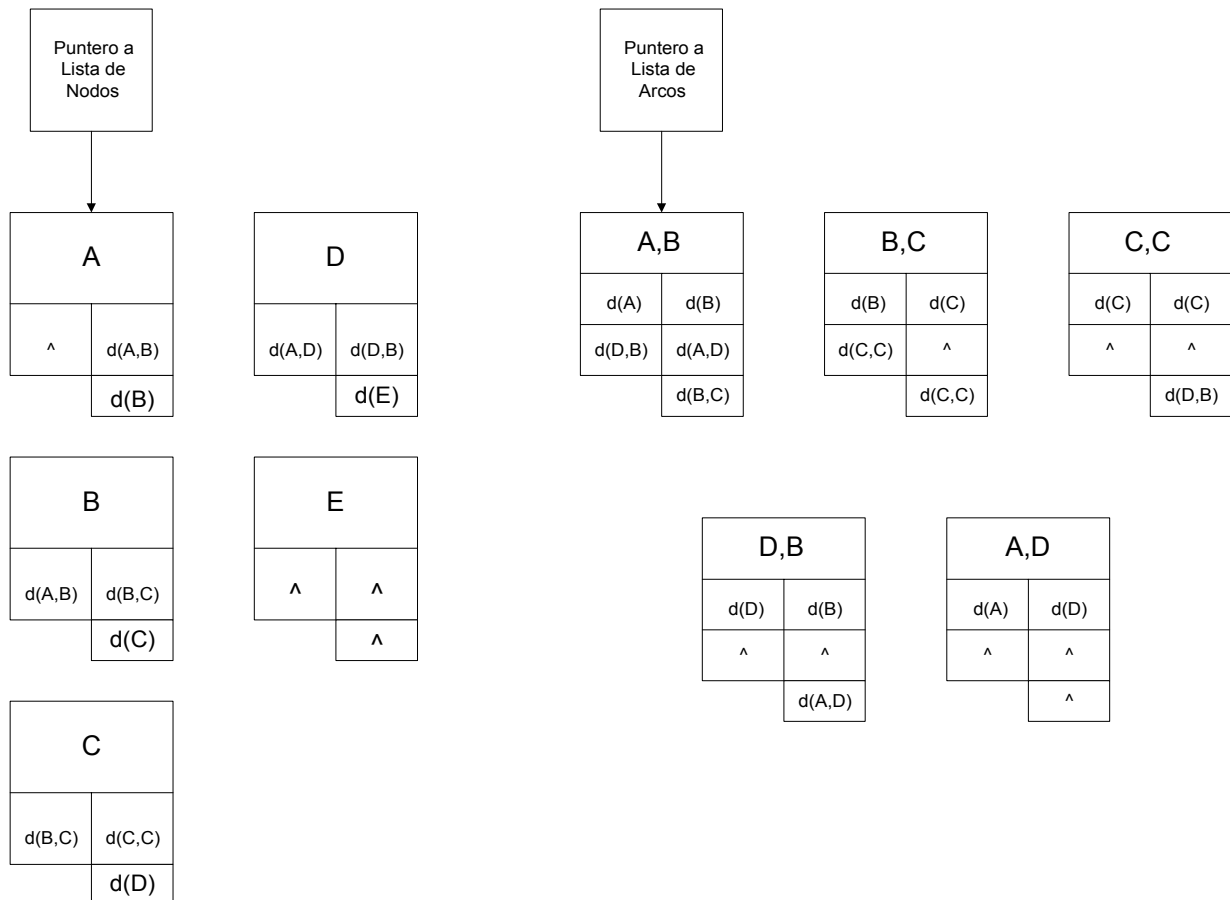
Ingreso de la relación (D,B)

- En la celda de tipo arco "C,C" se actualiza el next_a = d(D,B) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco "D,B" se actualizan lpoint = d(D) y rpoint = d(B)
- En la celda tipo nodo "D" se actualiza el redge = d(D,B) por ser el primer arco que parte de "C".
- En la celda tipo arco "A,B" se actualiza el llink = d(D,B) por ser este el próximo arco que llega a "B", o sea que tiene igual rpoint que "A,B".



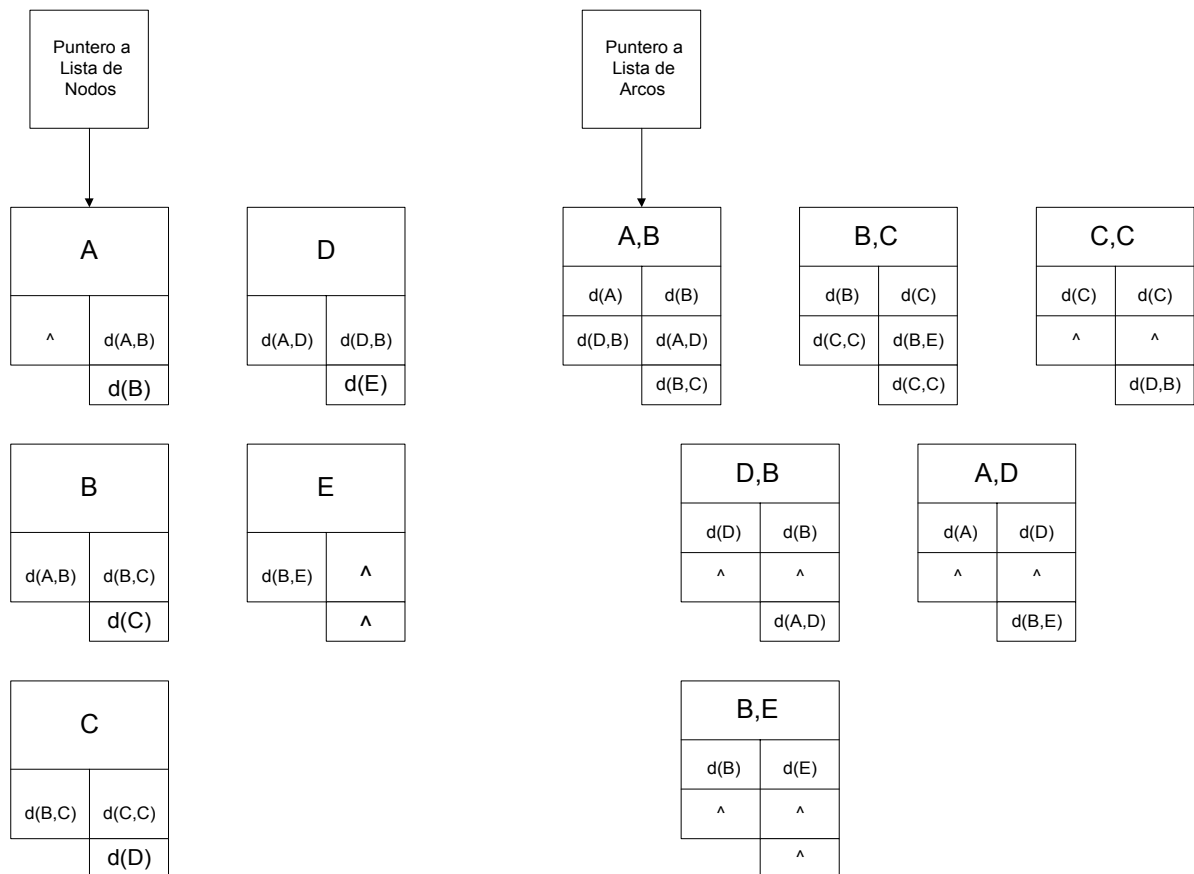
Ingreso de la relación (A,D)

- En la celda de tipo arco "B,D" se actualiza el next_a = d(A,D) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco "A,D" se actualizan lpoint = d(A) y rpoint = d(D)
- En la celda tipo nodo "D" se actualiza el ledge = d(A,D) por ser el primer arco que llega de "D".
- En la celda tipo arco "A,B" se actualiza el rlink = d(A,D) por ser este el próximo arco que sale de "A", o sea que tiene igual lpoint que "A,B".



Ingreso de la relación (B,E)

- En la celda de tipo arco "A,D" se actualiza el next_a = d(B,E) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco "B,E" se actualizan lpoint = d(B) y rpoint = d(E)
- En la celda tipo nodo "E" se actualiza el ledge = d(B,E) por ser el primer arco que llega a "E".
- En la celda tipo arco "B,C" se actualiza el rlink = d(B,E) por ser este el próximo arco que sale de "B", o sea que tiene igual lpoint que "B,C".



Ingreso de la relación (A,E)

- En la celda de tipo arco “B,E” se actualiza el next_a = d(A,E) por ser este el próximo arco de la lista de arcos.
- En la celda tipo arco “A,E” se actualizan lpoint = d(A) y rpoint = d(E)
- En la celda tipo arco “B,E” se actualiza el llink = d(A,E) por ser este el próximo arco que llega a “E”, o sea que tiene igual rpoint que “A,E”.
- En la celda tipo arco “A,D” se actualiza el rlink = d(A,E) por ser este el próximo arco que parte a “A”, o sea que tiene igual lpoint que “A,E”.

