rdo I OK

UNMDP	PROGRAMACION I	2º Parcial	04/11/2015
Matrícula: 39337854	Nombre: LAUTARO ECHEVERRÍA LUCHINI	NOTA :	8 (ocho)

Una empresa de transporte pretende gestionar información sobre las horas trabajadas por sus N choferes en una semana. Para ello cuenta con un archivo de texto con los siguientes datos:

- nombre del chofer y en las siguientes líneas (para cada día que trabajo):
    - Día (1..7, 0 indica fin)
    - Horas (siempre distinto de cero)

Se solicita almacenar dichos datos en estructuras adecuadas, que permitan obtener la siguiente información:

a) Día de la semana en la que hayan trabajado menos choferes (suponer único).

b) Generar un arreglo de registros con el nombre y el total de horas semanales de los choferes que hayan trabajado todos los días

c) A partir del arreglo del punto b) y del nombre de un chofer informar el total de horas semanales o cero si no se encuentra

<u>Nota</u>: al menos <u>dos</u> de los de los siguientes subprogramas (necesarios para resolver el problema planteado) deben ser <u>recursivos</u>.

- Iniciar en cero una matriz real de Nx7
- A partir de una matriz real de Nx7 y una columna determinada devolver la cantidad de ceros que contiene
- A partir de una matriz real de Nx7 y una fila determinada devolver TRUE si todos sus elementos son no nulos y FALSE en caso contrario
- A partir del vector descripto en b) y del nombre de un chofer devolver el total de horas trabajadas (0 de no encontrarlo).

<u>Lote de datos y resultados esperados</u>

Archivo texto
4
JUAN
1 4
2 7
6 9
0 0
RAMON
1 7
2 6
3 4
4 6
5 5
6 7
7 9
0 0

Archivo texto (cont.)
OSCAR
1 8
3 8
5 8
7 8
0 0
MARIA
1 9
2 4
3 9
4 4
5 9
6 9
7 4
0 0

Nombres
JUAN
RAMON
OSCAR
MARIA

Dias						
4	7				9	
7	6	4	6	5	7	9
8		8		8		8
9	4	9	4	9	9	4

a)
Dia 4

b)

Arreglo de registros	
Nombre	Hs. semanales
RAMON	44
MARIA	48

c)
MARIA = 48
OSCAR = 0

```
Program parcial;
Uses crt;
Type
ST10=string[10];
TM=array[1..20,1..7] of real;
TV1=array[1..20] of ST10;
TR=record
  nombre:ST10;
  horas:real;
  End;
TV2=array[1..20] of TR;

Procedure LeeMatriz(Var Mat:TM; Var N,M:byte; Var Nomb:TV1);
        Var
          Arch:text;
          i,j,dia:byte;
          horas:real;
        Begin
        M:=7; j:=0;
        Assign(Arch,'PARCIAL.txt');Reset(Arch);
        readln(Arch,N);
        For i:=1 to N do
          For j:=1 to M do
            Mat[i,j]:=0;
        For i:=1 to N do
          Begin
          readln(Arch,Nomb[i]);
          readln(Arch,dia,horas);
          While (dia>0) and (horas>0) do
            Begin
            Mat[i,dia]:=horas;
            readln(Arch,dia,horas);
            End;
          End;
        Close(Arch);
        End;

Function CantCeros(Mat:TM; i,M:byte):byte;
        Var
          suma:byte;
        Begin
        If i=1 then
          If Mat[i,M]=0 then
            CantCeros:=1
          Else
            CantCeros:=0
        Else
          Begin
          If Mat[i,M]=0 then
            suma:=1+CantCeros(Mat,i-1,M)
          Else
            suma:=0+CantCeros(Mat,i-1,M);
          CantCeros:=suma;
          End;
        End;

Function DiaMinChofer(Mat:TM; N,M:byte):byte;
        Var
          i,j,maxD:byte;
          Max,suma:real;
        Begin
        Max:=CantCeros(Mat,N,1);maxD:=1;
        For j:=2 to M do
```

debería ser una const

debería implementarlo como un procedimiento aparte.-

está de más

1

if i=0 then cantceros:=0

cantceros

cantceros

no

```
    Begin
    suma:=CantCeros(Mat,N,j);
    If Max<suma then
      Begin
      Max:=suma;
      maxD:=j;
      End;
    End;
  DiaMinChofer:=maxD;
  End;

Function TotalHoras(Mat:TM; N,M:byte):real;
  Var                                  No se usa
    i,j:byte;
    suma:real;
  Begin
  suma:=0;
  For j:=1 to M do
    suma:=suma+Mat[N,j];
  TotalHoras:=suma;
  End;

Function FilaSinCeros(Mat:TM; N,j:byte):boolean;
  Var
    suma:byte;   está de más
  Begin
  If j=1 then          1
    If Mat[N,j]=0 then              ≡ if j=0 then
      FilaSinCeros:=False                 FilasSinCeros := ~~false~~ true
    Else
      FilaSinCeros:=True
  Else                                  else
    Begin                                      if Mat[N,j] =0 then
    If Mat[N,j]=0 then                              FilasSinCeros := false
      suma:=1
    Else                                       else
      suma:=0;                                      FilasSinCeros:= FSceros(Mat,N
    If suma=1 then
      FilaSinCeros:=False
    Else
      FilaSinCeros:=FilaSinCeros(Mat,N,j-1);
    End;
  End;

Procedure ArregloB(Mat:TM; N,M:byte; Nomb:TV1; Var Reg:TV2; Var K:byte);
  Var      No VA
    i,j:byte;
  Begin
  K:=0;
  For i:=1 to N do                     está de más
    Begin
    If FilaSinCeros(Mat,i,M)=True then
      Begin
      K:=K+1;
      Reg[K].nombre:=Nomb[i];
      Reg[K].horas:=TotalHoras(Mat,i,M);
      End;
    End;
  End;

Procedure Busqueda(Reg:TV2; K:byte);
  Var
    i:byte;
```

```
    nombX:ST10;
  Begin
  i:=1;
  writeln('Ingresar el nombre del chofer.');
  readln(nombX);
  While (i<=K) and (Reg[i].nombre<>nombX) do
    i:=i+1;
  If i>K then
    writeln('0')
  Else
    writeln('El total de horas semanales que trabajo fueron:
',Reg[i].horas:4:2);
  End;

Procedure Muestra(Mat:TM; N,M:byte; Nomb:TV1);
  Var
    i,j:byte;
  Begin
  writeln('La matriz es:');
  For i:=1 to N do
    Begin
    For j:=1 to M do
      write(Mat[i,j]:4:2,'   ');
    writeln;
    End;
  writeln('El vector es:');
  For i:=1 to N do
    writeln(Nomb[i]);
  End;

Procedure MuestraArregloB(Reg:TV2; K:byte);
  Var
    i:byte;
  Begin
  writeln('Los choferes que trabajaron todos los dias son:');
  For i:=1 to K do
    Begin
    write(Reg[i].nombre,'   ',Reg[i].horas:4:2);
    writeln;
    End;
  End;

Var
 Mat:TM; N,M,K:byte;
 Nomb:TV1; Reg:TV2;

Begin
  clrscr;
  LeeMatriz(Mat,N,M,Nomb);
  Muestra(Mat,N,M,Nomb);
  writeln('El dia que menos choferes trabajon es: ',DiaMinChofer(Mat,N,M));
  ArregloB(Mat,N,M,Nomb,Reg,K);
  MuestraArregloB(Reg,K);
  Busqueda(Reg,K);
  readln;
End.
```

Podría ser una fción q/ devuelva ∅ o la posición

FECHA: / /

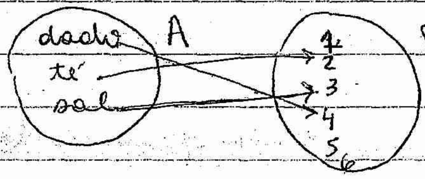
UNIDAD 3: RELACIONES ENTRE CONJUNTOS.

Relación BINARIA.

Sean A y B conjuntos. Una relación binaria, o simplemente una relación de A a B, es un subconjunto del producto cartesiano A x B.

Representación de relaciones binarias definidas sobre conjuntos finitos

DIAGRAMA SAGITAL



MATRIZ DE ADYACENCIA.

$$M_R = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$M_R = [m_{ij}]$$

$$m_{ij} = \begin{cases} 1 & \text{si } a_i R b_j \\ 0 & \text{en caso contrario} \end{cases}$$

Definición: Sean A y B conjuntos entre los que está definida ~~una relación~~ R, llamada R^{-1} es un subconjunto de B x A, que cumple que

$$R^{-1} = \{(b,a) / (a,b) \in R\}$$

$$M_{R^{-1}} = (M_R)^t$$

Definición: PRODUCTO BOOLEANO DE MATRICES.

Si A es una matriz booleana de f x c elementos y B es una matriz booleana que tiene c x r elementos, la matriz $A \odot B$ tiene f x r elementos obtenidos

elemento $(a \odot b)_{ij} = a_{i1} \wedge b_{1j} \vee a_{i2} \wedge b_{2j} \vee \dots$
$a_{ic} \wedge b_{cj}$

$$M_{S \circ R} = M_R \odot M_S$$

Definición: Precedencia de matrices booleanas.

Si A y B son matrices booleanas de $f \times c$ elementos, se dice que A precede a la matriz B.

Se denota $A \leq B$ sii $a_{ij} \leq b_{ij}$ $\forall\, 1 \leq i \leq f$, $\forall\, 1 \leq j \leq c$

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

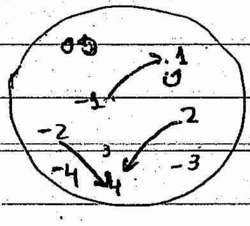
- A precede a B pero no a C.
- C precede a B.

Relaciones binarias en un único conjunto.

Sea A un conjunto. Una relación binaria, o simplemente una relación en A, es un subconjunto de $A \times A$.

Ej $A = \{0, 1, -1, 2, -2, -3; 3; 4; -4\}$

$$R = \{(x,y) \in A^2 / x^2 = y\} = \{(-1,1), (1;1), (2,4), (-2,4), (0,0)\}$$



$\Rightarrow$ El diagrama sagital recibe el nombre de DIGRAFO de la relación.

Operaciones con relaciones.

Dadas R_1 y R_2 definidas sobre A.

- Unión: $R_1 \cup R_2 = \{(a,b) \in A \times A \,/\, a R_1 b \text{ ó } a R_2 b\}$

- Composición o producto: $R_2 \circ R_1 = \{(a,b) \in A \times A \,/\, \exists c \in A \;\; a R_1 c \wedge c R_2 b\}$

- En general, $R_1 \circ R_2 \neq R_2 \circ R_1$
- La composición es asociativa: $R^{n+1} = R \circ R^n$

Operaciones con relaciones.

Dada R sobre $A = \{a_1, \dots, a_n\}$ y M_R su adyacencia:

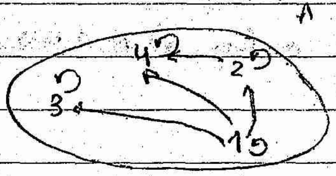
- $M_R = O_R \Longleftrightarrow R = \emptyset$ (matriz nula de orden n.)
- $M_R = 1_R \Longleftrightarrow R = A \times A$ (matriz de unos de orden n.)
- $M_R^m = (\odot M_R)^m$, $m \in \mathbb{N}$ (m-ésima potencia booleana).

Propiedades: para R definida sobre A, con matriz de adyacencia M y $|A| = n$, se consideran:

1) Reflexividad:

R es reflexiva sii $\forall\, x \in A$ se verifica que $(x,x) \in R$.

Ej: $A = \{1,2,3,4\}$ con la $x R y$ sii $x | y$



Bucles en el dígrafo.

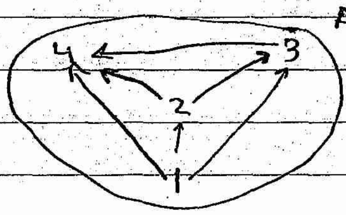
$$M_R = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

R es reflexiva.

1's en la diagonal de la matriz, es decir $Id_A \leq M_R$

II) Arreflexividad: R es arreflexiva sii $\forall\, x \in A$ se verifica que $(x,x) \notin R$.

Ej: $A = \{1,2,3,4\}$ con la relación $x R y$ sii $x < y$



No hay bucles en el dígrafo.

$$M_R \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

0's en la diagonal principal.

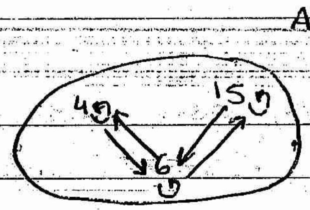
R es arreflexiva.

III) Simetría:

R es simétrica sii $\forall x, y \in A$ se verifica que:

$$(x,y) \in R \Longrightarrow (y,x) \in R$$

Ej: $A = \{4, 6, 15\}$
con la relación xRy sii $d = (x;y) > 1$



$$M_R = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

M_R es simétrica, es decir $M_R = M_R$ traspuesta.

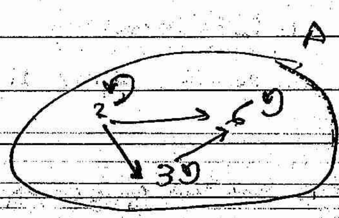
Flechas ida y vuelta en el digrafo.

R es simétrica.

IV) Antisimétrica:

R es antisimétrica sii $\forall x, y \in A$ se verifica que: $(x,y) \in R \wedge (y,x) \in R \Rightarrow x$

ejemplo: $A = \{2, 3, 6\}$
con la relación xRy sii $x | y$



$$M_R = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

No hay flechas ida y vuelta, salvo bucles...

$M_R \wedge M_R^t$ precede a la Id_n

R es antisimétrica.

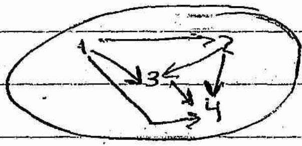
FECHA: / /

V) Asimetría:

R es asimétrica sii $x, y \in A$ se verifica que: $(x,y) \in R \rightarrow (y,x) \notin R$.

Ej: $A = \{1,2,3,4\}$ con la relación xRy sii $x < y$

A



$$M_R \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

No hay bucles ni flechas ida y vuelta en digrafo.

$M_R \wedge M_R^t = O_n$

R es asimétrico.

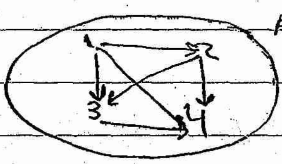
VI) Transitividad:

R es transitiva sii $\forall\, x, y, z \in A$ se verifica que: $(x,y) \in R \wedge (y,z) \in R \rightarrow (x,z) \in R$

Ej: $A = \{1,2,3,4\}$ con la relación xRy sii $x < y$

$$M_R = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

A



Si hay sucesión de 2 flechas, hay "ATAJO".

M_{R^2} precede M_R.

R es transitivo.

Relación de equivalencia:

• Para R definido sobre A, se dice que R es una relación de equivalencia sii R es reflexiva, simétrica y transitiva.

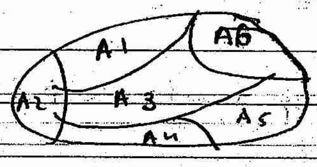
→ A es un conjunto y R es una relación de equivalencia de A. Para cada elemento a en A, la clase de equivalencia de a, que se denota [a] y se llama la clase de a, es el conjunto de todos los elementos x en A tales que x está relacionado con a por R

$\Rightarrow [a] = \{x \in A$ tales que $xRa\}$

• Particiones y relaciones de equivalencia:

• dado un conjunto A, una partición de A es una colección finita o infinita de conjuntos A_i que cumplen que

$A_i \cap A_j = \emptyset$ para $i \neq j$ y $\bigcup_{i=1}^{n} A_i = A$



$A_i \cap A_j = \emptyset$ siempre que $i \neq j$

$A_1 \cup A_2 \cup \ldots \cup A_6 = A$

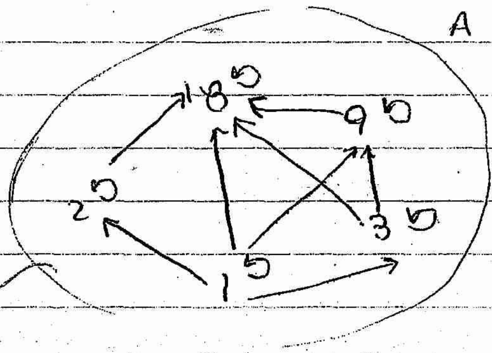
CONJUNTO COCIENTE $A/R = \{[a]$ con $a \in A\}$

Idea central del TEOREMA FUNDAMENTAL DE LAS RELACIONES DE EQUIVALENCIA: Las clases de equivalencia determinan una partición sobre A.
Recíprocamente, se puede definir una relación de equivalencia en un conjunto A inducida por una partición

Relaciones de orden parcial

Para R definida sobre A, se dice que R es una relación de orden parcial si R es Reflexiva, antisimétrica y transitiva.

ej: $A = \{1, 2, 3, 9, 18\}$ $x R y$ sii $x \mid y$



En un orden parcial si aRb se dice que a "precede a" b.

DIAGRAMA DE HASSE

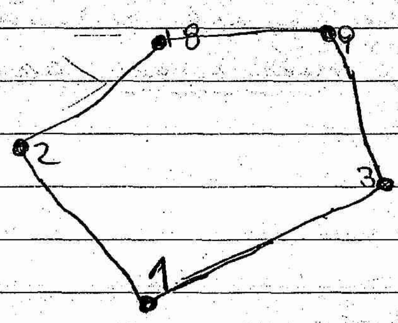
Construcción del diagrama:

1º) Se omiten bucles

2º) Se eliminan los aristas que tienen que ser presentes por la transitividad ("atajos").

3º) Se ubica cada arista de manera que su vértice inicial quede por debajo de su vértice final y se eliminan las indicaciones de sentido de las flechas.

DIAGRAMA de HASSE.



- Elementos comparables:

definición: Dada una relación $\leq$ de orden parcial sobre un conjunto A. Se dice que los elementos a y b de A son comparables si y solo si, ya sea $a \leq b$ ó $b \leq a$. De otra manera, a y b no son comparables.

ej: $2 \nmid 9$ y $9 \nmid 2$ no son comparables por el orden parcial R.

• Para todo par de elementos en B se cumple que son comparables por el orden parcial R. R es un orden TOTAL o lineal.

Ej: $B = \{A, B, C, D, E\}$

$x R y$ sii (x coincide con y) $\vee$ (x es anterior en el alfabeto a y).

DIAGRAMA de HASSE.

Def: dada una relación $\leq$ de orden parcial sobre un conjunto A, si para cualquier par de elementos a y b de A se verifica que son comparables ($a \leq b$ o $b \leq a$), se dice que es un orden total o lineal. En ese caso se dice que A está TOTALMENTE ORDENADO por la relación $\leq$.

SUBCONJUNTOS CADENAS.

Un subconjunto B de A es una cadena si y solo si cada par de elementos en B es comparable. En otras palabras, $a \leq b$ o $b \leq a$ para todos a y b en B

ej: $A = \{1, 2, 3, 4, 18\}$ $x R y$ sii $x | y$

Elementos notables

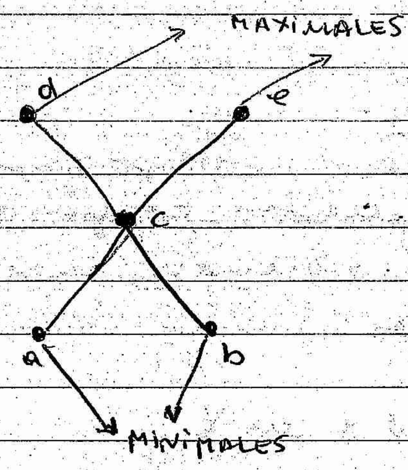
• Maximales, minimales, máximos y mínimos.

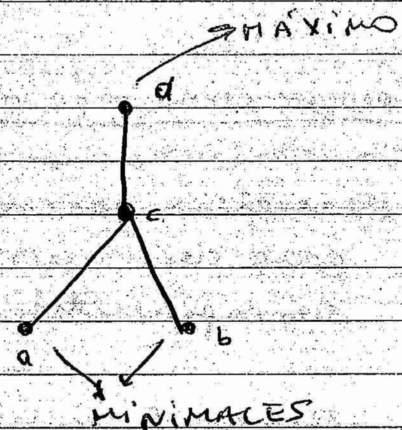
• Un elemento a en A se llama un elemento maximal de A sii $\forall x \in A \ (x \leq a) \vee (x$ y a No son comparables$)$.

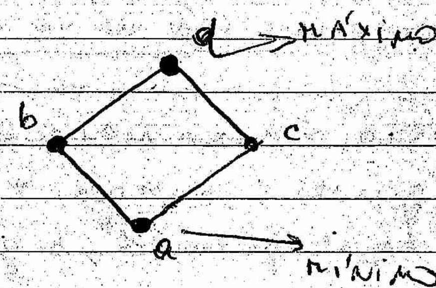
• Un elemento a en A se llama máximo de A sii $\forall x \in A$ $x \leq a$.

• Un elemento a en A se llama un elemento minimal de A sii: $\forall x \in A \ (a \leq x) \vee (x$ y a No son comparables$)$

• Un elemento a en A se llama mínimo de A sii $\forall x \in A$ $a \leq x$





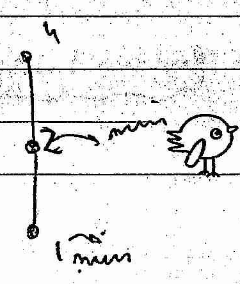


cualquier conjunto parcialmente ordenado que es finito y no vacío tiene minimal (maximal)

Def. $(A, \leq)$ es un conjunto bien ordenado si $\leq$ es un orden total sobre un conjunto A y cualquier subconjunto A y cualquier subconjunto vacío de A tiene un elemento mínimo

Ej: $A = \{1, 2, 4\}$ $x \leq y$ sii $x | y$

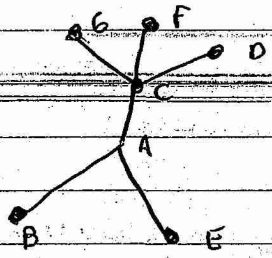
A está bien ordenado



⇒ De un orden parcial a un orden total:

ORDEN TOPOLÓGICO

Hay un orden parcial $\leq$ definido por el diagrama de H



Se puede construir un ORDEN TOTAL $\leq'$ que conserve el orden parcial del D. de H., es decir, que $\leq'$ sea compatible con el orden $\leq$.
A un nuevo orden total $\leq'$ se le llama ORDENAMIENTO TOPOLÓGICO para $\leq$

Def. Dadas las relaciones de orden parcial sobre un conjunto A, $\leq$ y $\leq'$, se dice que $\leq'$ es un ordenamiento topológico para $\leq$, sii $\leq'$ es un orden total compatible con $\leq$.

• CONSTRUCCIÓN DE UN ORDENAMIENTO TOPOLÓGICO.

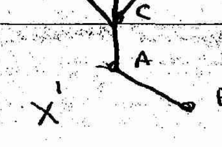
Sea $\leq$ una relación de orden parcial sobre un conjunto finito no vacío X. Para construir un ordenamiento topológico $\leq'$, siga los siguientes pasos:

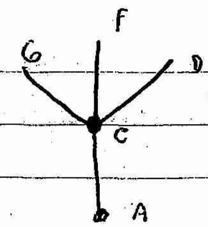
1) Elija cualquier elemento minimal x en X.
2) Sea $X' := X - \{x\}$.
3) Repita los pasos a-c en tanto $X' \neq \emptyset$.

   a. Elija cualquier elemento minimal y en X'
   b. Defina $x \leq' y$,
   c. Sea $X' := X' - \{y\}$ y $x := y$

Para el ejemplo: Elegimos uno de los minimales (B)

$X' = X - \{B\}$,

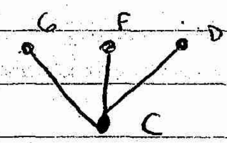




Elegimos el minimal E en X'

Definimos $B \leq' E$

Sea $X' := X' - \{E\}$ y $x := E$



Elegimos el minimal A en X'

Definimos $E \leq' A$

Sea $X' := X' - \{A\}$ y $x := A$

y así para C luego G, F y D.

• Hasta que finalmente $X' = \emptyset$ y obtengo:

$$B \leq' E \leq' A \leq' C \leq' G \leq' F \leq' D$$

Cotas para subconjuntos de un conjunto parcialmente ordenado.

Dada una relación $\leq$ de orden parcial sobre un conjunto A y $S \subseteq A$.

1) Un elemento a en A se llama COTA INFERIOR sii $\forall x \in S$ $a \leq x$

2) Un elemento a en A se llama ÍNFIMO DE S sii a es cota inferior de S y $\forall a' \in A$ a' cota inferior de S $a' \leq a$.

3) Un elemento a en A se llama cota SUPERIOR de S sii $\forall x \in S$ $x \leq a$

4) Un elemento a en A se llama SUPREMO DE S sii a es cota superior de S y $\forall a' \in A$, a' cota superior de S $a \leq a'$.

~~Def. Dada una relación ≤ de orden parcial sobre un conjunto A y S ⊆ A~~

Sucesiones

• definidas por relaciones de recurrencia:

Las relaciones de recurrencia se usan para analizar el tiempo que requieren los algoritmos. La técnica consiste en desarrollar una relación de recurrencia y condiciones iniciales que definan una sucesión $a_1, a_2, \ldots$, donde a_n es el tiempo (en el mejor caso, caso promedio o peor caso) requerido por un algoritmo para ejecutar una entrada de tamaño n. Al resolver la relación de recurrencia, con la fórmula explícita de la sucesión se determinará el tiempo que requiere el algoritmo.

Definición: Una relación de recurrencia para una sucesión $a_0, a_1, a_2 \ldots$ es una fórmula que relaciona cada término a_k con algunos de sus predecesores $a_{k-1}, a_{k-2}, \ldots, a_{k-i}$ donde i es un número entero con $k-i \geq 0$. Las condiciones iniciales para una relación de recurrencia especifican los valores de $a_0, a_1, a_2, \ldots, a_{i-1}$ si i es un entero fijo, o $a_0, a_1, \ldots, a_m$ donde m es un número entero con $m \geq 0$, si i depende de k.

Ej: sucesión de Fibonacci $F_k = F_{k-1} + F_{k-2}$.

condiciones iniciales $F_0 = 1$ $F_1 = 1$

Solución de Relaciones recursivas

• Resolver una relación de recurrencia que implica una sucesión $a_0, a_1, \dots$ significa encontrar una fórmula explícita para el término general a_n.
Veremos dos métodos para resolver relaciones de recurrencia: el de iteraciones y un método especial que se aplica a relaciones de recurrencia homogéneas lineales con coefs. ctes.
Hay otros métodos más poderosos, que no veremos.

Solución por Iteración:

Método: dada una sucesión $a_0, a_1, a_2, \dots$, definida por una relación de recurrencia y condiciones iniciales, inicie a partir de las ci y calcule los términos sucesivos de la sucesión hasta que aparezca un patrón de desarrollo. En ese momento, proponga una fórmula explícita.

Una sucesión $a_0, a_1, a_2 \dots$ se llama sucesión aritmética, si existe una cte d /

$$\underline{a_k = a_{k-1} + d}, \text{ para todo entero } k \geq 1$$

De lo que se tiene que,

$$\underline{a_n = a_0 + d \cdot n}, \text{ para todo entero } n \geq 0$$

Una sucesión $a_0, a_1, a_2 \dots$ se llama sucesión geométrica, si existe una cte r /

$$\underline{a_k = r \cdot a_{k-1}}, \text{ para todo entero } k \geq 1$$

De lo que se tiene que,

$$\underline{a_n = a_0 \cdot r^n}, \text{ para todo entero } n \geq 0.$$

Soluciones por relaciones de recurrencia homogéneas de 2do orden con coefs. ctes.

es de la forma $a_k = A a_{k-1} + B \cdot a_{k-2}$, para todo entero $k \geq$ que algún entero fijo, donde A y B son números reales fijos con $B \neq 0$.

LEMA 1:

Sean A y B números reales. Una relación de recurrencia de la forma

$$\boxed{a_k = A \cdot a_{k-1} + B \cdot a_{k-2}}$$

para todo entero $k \geq 2$

se satisface con la sucesión

$$1, t, t^2, t^3, \dots, t^n, \dots,$$

donde t es un número real $\neq 0$, si t satisface la

$$\underbrace{t^2 - A\,t - B = 0}$$

→ Ec característica.

LEMA 2:

Si $r_0, r_1, r_2, \dots$ y $s_0, s_1, s_2, \dots$ son sucesiones que satisfacen la misma relación de recurrencia lineal de 2do orden lineal homogénea con coefs. ctes y si C y D son cualesquiera números, entonces la sucesión $a_0, a_1, a_2, \dots$ definida por la fórmula

$$\boxed{a_n = C r_n + D s_n}$$ para todo entero n

también satisface esa misma relación de recurrencia

FECHA: / /

LEMA 3:

Supongamos que una sucesión $a_0, a_1, a_2, \ldots$, satisface una relación de recurrencia

$$a_k = A \cdot a_{k-1} + B \cdot a_{k-2}$$

para algunos números reales A y B con $B \neq 0$ y para todos los enteros $k \geq 2$.

Si la ec. característica $t^2 - At - B = 0$, tiene una única raíz r (real), entonces, $a_0, a_1, a_2 \ldots$ está dada por la fórmula explícita

$$\boxed{a_n = C r^n + D \cdot n \cdot r^n}$$

donde C y D son los números reales cuyos valores se determinan por los valores de a_0 y de cualquier otro valor conocido de la sucesión.