MATEMÁTICAS IV (Matemáticas Discretas)

Objetivo General

El alumno aplicara la teoría de las Matemáticas Discretas en la interpretación y resolución de problemas algorítmicos, gráficas, inducción y recursion.

Contenido

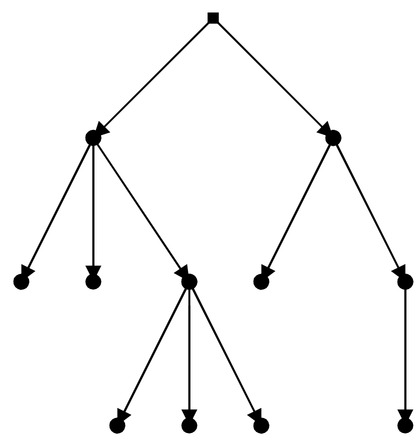
1. Inducción

1.1. Objetivo particular

El objetivo del presente tema es el de introducir al alumno al ámbito de las graficas, caminos, árboles , así como a como de las matrices para graficas.

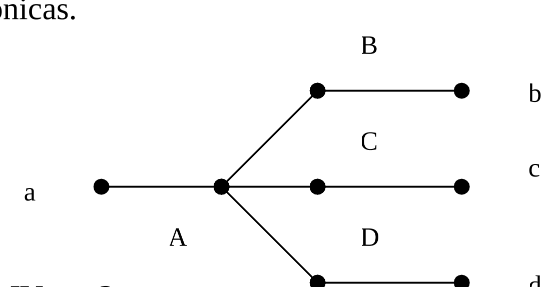
1.2. Gráficas

Una grafica son objetos y líneas. Los objetos están representados por puntos, se llaman vértices y las líneas que los unen aristas.
Un camino son las aristas que se unen.
Ejemplo de una grafica, lo tenemos en seguida:



1.3. Caminos y Árboles especiales

Los árboles son una clase de grafos. Un claro ejemplo de un árbol es el siguiente: consideremos cuatro parejas de chismosos {a, A, b, B, c, C, d, D}, donde a, b, c, y d son los esposos y A, B, C, y D, son, respectivamente, sus esposas. Supongamos que a llama a su esposa para contarle algún chisme, entonces ella llama a las otras esposas para difundir el chisme, y cada una de ellas, a su vez, llama a su esposo para comunicárselo. El grafo en la fig.1 muestra cómo se propagó el chisme, y dónde las aristas representan las llamadas telefónicas.



Un árbol es un grafo (no dirigido) conexo que no contiene circuitos, es decir, que no existen dos o más paseos entre un par de vértices.

Una colección de árboles disjuntos es llamado un bosque. Un vértice de grado 1 en un árbol se llama una hoja o un nodo terminal, y un vértice de grado mayor que 1 recibe el nombre de un nodo rama o nodo interno. Por ejemplo, son hojas b, c, d y los vértices a, A, B, C, D son nodos rama.

Las propiedades de los árboles son:

- Existe un único paseo entre dos vértices cualesquiera en un árbol.
- El número de vértices es mayor en uno al número de aristas en un árbol.
- Un árbol con dos o más vértices tiene al menos dos hojas.

Un grafo dirigido es un conjunto de puntos marcados V con un conjunto de flechas entre los puntos, de manera que a lo más existe una flecha desde un punto a otro punto.

Un grafo no dirigido es un conjunto de puntos marcados V con un conjunto de líneas E entre los puntos.

Un grafo no dirigido es conexo si existe un paseo entre cualesquiera dos vértices, y es no conexo en otro caso. Se dice que un grafo dirigido es conexo si el grafo no dirigido derivado de éste, al ignorar las direcciones de las aristas, es conexo y es no conexo en otro caso. Entonces se tiene que un grafo no conexo consiste en dos o más componentes, cada una de las cuales es un grafo conexo.

1.4. Matrices para gráficas

Se designa con el nombre de técnicas matriciales a la representación cruzada de diferentes entidades u objetos de interés para la Organización y que permiten:

- Conocer la realidad actual en cuanto a sus funciones, información manejada, distribución geográfica, etc.

- Sentar las bases para una posible reorganización de las funciones con objeto de aumentar su eficacia.

- Definir nuevos sistemas de información para la Organización.
- Ayudar a definir prioridades en el desarrollo de nuevos sistemas.

Las diferentes representaciones matriciales que se recogen en la metodología son las siguientes:

- **Matriz Procesos-Entidades de Datos:** que permite representar el tratamiento lógico de las funciones sobre los datos del sistema.
- **Matriz Procesos-Organización:** que permite representar tanto la distribución geográfica de las funciones de la Organización, como las responsabilidades de los distintos departamentos en que se estructura.
- **Matriz Aplicaciones-Ficheros de Datos:** que permite conocer la situación actual de los sistemas de información existentes, en cuanto a las aplicaciones en funcionamiento y los datos que manejan.
- **Matriz Aplicaciones-Funciones:** que permite representar el grado de cobertura que las aplicaciones existentes tienen sobre las funciones que desarrolla, o tiene previsto desarrollar la Organización.
- **Matriz Ficheros de Datos Actuales-Entidades de Datos:** que permite representar la adecuación de los ficheros de datos existentes a las necesidades de información de la Organización.

Como se puede observar en este conjunto de matrices, existe un grupo cuyo objetivo básico es realizar el estudio detallado de la situación actual y el grado de satisfacción de las necesidades de información de la Organización con los sistemas existentes formado por:

- **Matriz Aplicaciones-Ficheros de Datos.**
- **Matriz Aplicaciones-Funciones.**
- **Matriz Ficheros de Datos actuales-Entidades de Datos.**

Matriz de adyacencia: Matriz de dimensión n x n cuyos elementos hijos toman los valores:

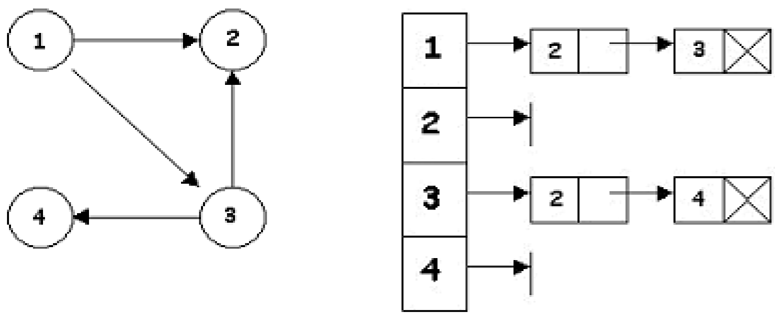
0 si no existe el arco i,j

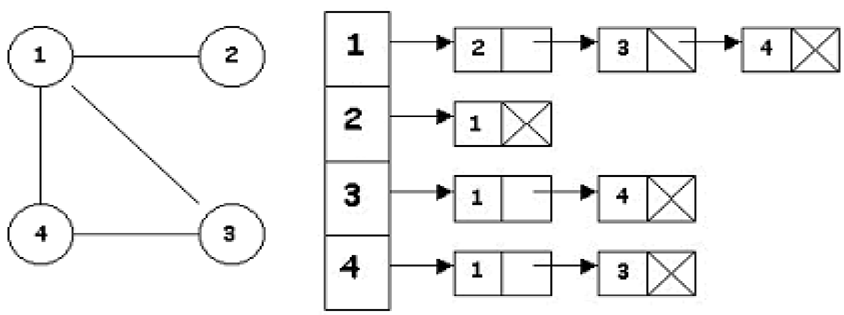
1 si existe el arco i,j

- En las columnas se encuentran los nodos finales y en las filas los nodos iniciales.
- La sumatoria de cada fila los da los grados de salida de cada nodo
- La sumatoria de cada columna nos da los grados de entrada de cada nodo

La representación de la matriz de adyacencia es útil para aquellos algoritmos que precisan saber si existe o no una arista entre dos vértices dados.

El grafo, por tanto, se representa por medio de un vector de n componentes (si $|V| = n$) donde cada componente va a ser una lista de adyacencia correspondiente a cada uno de los vértices del grafo. Cada elemento de la lista consta de un campo indicando el vértice





adyacente.

En caso de que el grafo sea etiquetado, habrá que añadir un segundo campo para mostrar el valor de la etiqueta.

Esta representación requiere un espacio proporcional a la suma del número de vértices, más el número de arcos, y se suele usar cuando el número de arcos es mucho menor que el número de arcos de un grafo completo. Una desventaja es que puede llevar un tiempo O(n) determinar si existe un arco del vértice *i* al vértice *j*, ya que pueden haber n vértices en la lista de adyacencia asociada al vértice *i*.

Más aún, para saber, por ejemplo, si un grafo no dirigido representado mediante su matriz de adyacencia es conexo, o simplemente para conocer el número de aristas, los algoritmos requieren un tiempo en (n2), lo cual es más de lo que cabría esperar.

La representación de un grafo con listas de adyacencia requiere un espacio del orden del máximo entre n (el número de vértices) y a (el número de aristas).

Una ventaja de esta representación es que es útil para implementar aquellos algoritmos que necesitan recorrer un grafo y al mismo tiempo "marcar" las aristas por las que se pasa.

2. Computabilidad y lenguajes formales.

2.1. Objetivo particular

El objetivo del presente tema es el de ver la teoría de conjuntos sus operaciones, así como el manejo de subíndices e índices, y como realizar demostraciones formales.

2.2. Conjuntos especiales

2.2.1. CONJUNTO VACÍO

Es un conjunto que carece de elementos. Se suele llamarle conjunto nulo, y se le denota por el símbolo ø o { }.

Ejemplo:

A = { Los perros que vuelan }	A = { }	A = **Ø**
B = { x / x es un mes que tiene 53 días}	B = { }	B = **Ø**
C = { x / x^3 = 8 y x es impar }	C = { }	C = **Ø**
D = { x / x es un día de 90 horas }	D = { }	D = **Ø**

2.2.2. CONJUNTO UNITARIO

Es todo conjunto que está formado por un sólo y único elemento.

Ejemplo:
A = { 5 }
B = {números pares entre 6 y 10} = { 8 }
C = {la capital del Perú } = { Lima }
D = {x / 2x = 6} = {3}

2.2.3. CONJUNTO POTENCIA

La familia de todos los subconjuntos de un conjunto M se llama Conjunto Potencia de M. Se le denota como 2^M.
Ejemplo:

a) M = { 1, 2 } — El conjunto M tiene 2 elementos

2^M = { {1}, {2}, {1, 2}, ø} — entonces 2^2 = 4 elementos

b) M = { 1, 2, 3 } — El conjunto M tiene 3 elementos

2^M = { {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}, ø} — entonces 2^3 = 8 elementos

Si un conjunto M es finito con "n" elementos, entonces su conjunto potencia 2^M tendrá 2^n elementos.

2.3. Operaciones de conjuntos

2.3.1. UNIÓN DE CONJUNTOS

La unión de los conjuntos A y B es el conjunto formado por todos los elementos que pertenecen a A o a B o a ambos. Se denota: A U B. La unión de conjuntos se define como:

A U B = {x / x $\in$ A o x $\in$ B}

En forma gráfica:

Cuando no tienen elementos comunes	Cuando tienen algunos elementos comunes	Cuando todos los elementos de un conjunto pertenecen a otro conjunto

Ejemplo:

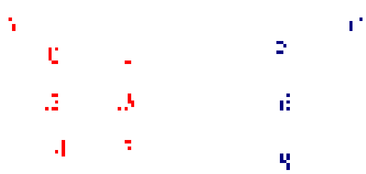
1. Dados los conjuntos: A = { 0, 1, 2, 3, 4, 5 }, B = { 0, 2, 4 } y C = { 5, 6, 8 }, efectuar y construir los diagramas respectivos:

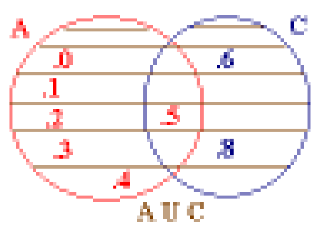
a) A U C b) B U C c) A U B

Tenemos:

a) A = { 0, 1, 2, 3, 4, 5 } y C = { 5, 6, 8 }

A U C = { 0, 1, 2, 3, 4, 5, 6, 8 }

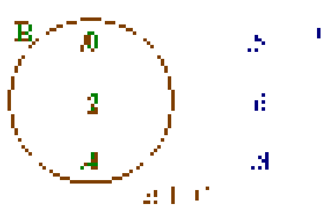




Representación gráfica de la unión de conjuntos A y C

b) B = { 0, 2, 4 } **y** C = { 5, 6, 8 }

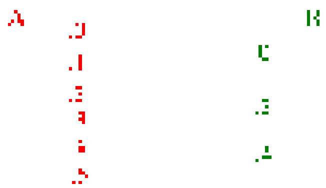
B U C = { 0, 2, 4, 5, 6, 8 }

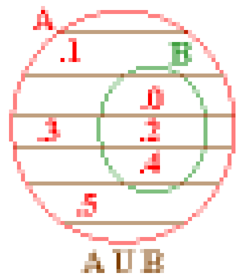


Representación gráfica de la unión de conjuntos B y C

c) A = { 0, 1, 2, 3, 4, 5 } y B = { 0, 2, 4 }

A U B = { 0, 1, 2, 3, 4, 5 }

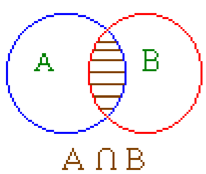


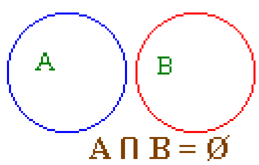


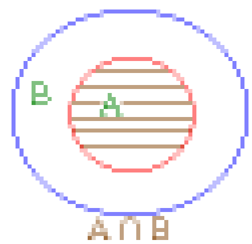
Representación gráfica de la unión de conjuntos A y B

2.3.2. INTERSECCIÓN DE CONJUNTO

Se define la intersección de dos conjuntos A y B al conjunto de elementos que son comunes a A y B. Se denota por $A \cap B$, que se lee: A intersección B. La intersección de A y B también se puede definir:
$A \cap B = \{ x / x \in A \text{ y } x \in B \}$ y mediante un diagrama de Venn-Euler:







Cuando tienen elementos comunes	Cuando no tienen elementos comunes	Cuando todos los elementos de un conjunto pertenecen a otro conjunto

Ejemplo:

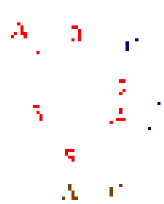
1. Dados los conjuntos: A = { 0, 1, 2, 3, 4, 5 }, B = { 3, 5, 7 } y C = { 2, 4 }, efectuar y construir los diagramas respectivos:

a) $A \cap C$ b) $B \cap C$ c) $A \cap B$

Tenemos:

a) A = { 0, 1, 2, 3, 4, 5 } y C = { 2, 4 }

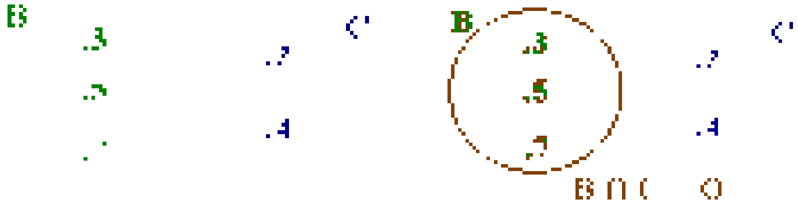
$A \cap C$ = { **2**, **4**}



Representación gráfica de la intersección de conjuntos A y C

b) B = { 3, 5, 7 } **y** C = { 2, 4 }

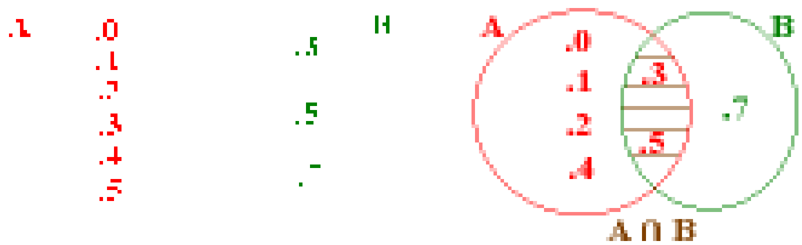
$B \cap C = \{ \}$



Representación gráfica de la intersección de conjuntos B y C

c) A = { 0, 1, 2, 3, 4, 5 } y B = { 3, 5, 7 }

$A \cap B = \{ 3, 5 \}$



Representación gráfica de la intersección de conjuntos A y B

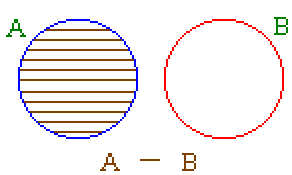
2.3.3. DIFERENCIA DE CONJUNTOS

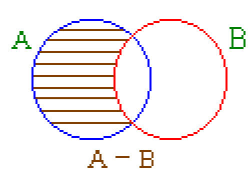
Se denomina diferencia de dos conjuntos A y B al conjunto formado por todos los elementos de A pero que no pertenecen a B.

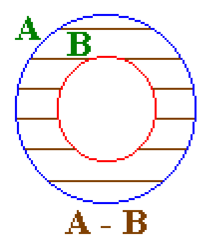
La diferencia se denota por: A - B que se lee: A diferencia B o A menos B. Se define la diferencia de dos conjuntos también como:

$A - B = \{x \,/\, x \in A \text{ y } x \notin B\}$

Mediante un diagrama de Venn - Euler:







Cuando no tienen elementos comunes

Cuando tienen elementos comunes

Cuando todos los elementos de un conjunto pertenecen a otro conjunto

Ejemplo:

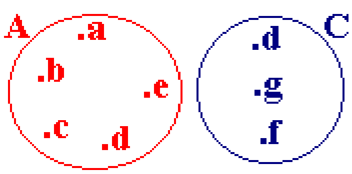
1. Dados los conjuntos: A = { a, b, c, d, e }, B = { a, e } y C = { d, f, g }, efectuar y construir los diagramas respectivos:

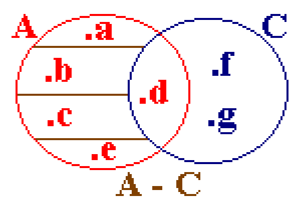
a) A – C b) B - C c) A - B

Tenemos:

a) A = { a, b, c, d, e } y C = { d, f, g }

A - C = { a, b, c, e }

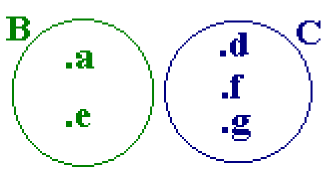


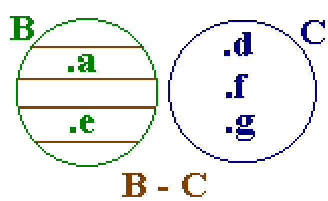


Representación gráfica de la diferencia de conjuntos A y C

b) B = { a, e } y C = { d, f, g }

B - C = { a, e }

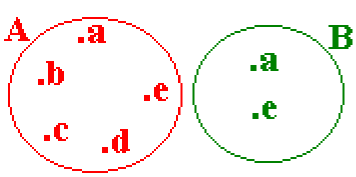


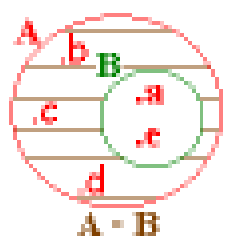


Representación gráfica de la diferencia de conjuntos B y C

c) A = { a, b, c, d, e } y B = { a, e }

A - B = { b, c, d }





Representación gráfica de la diferencia de conjuntos A y B

2.3.4. COMPLEMENTO DE UN CONJUNTO

Si un conjunto A es subconjunto de otro conjunto universal U, *al conjunto A*' formado por todos los elementos de U pero no de A, se llama *complemento de A* con respecto a U. Simbólicamente se expresa:

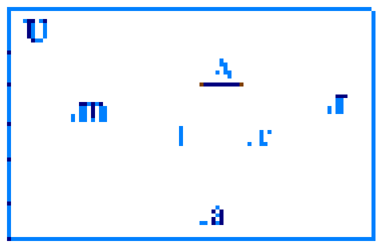
A' = { x/x $\in$ U y x $\notin$ A }

Ejemplo:

a) Sean U = { m, a, r, t, e } Y A = { t, e }

Su complemento de A es: A' = { m, a, r }

En forma gráfica:



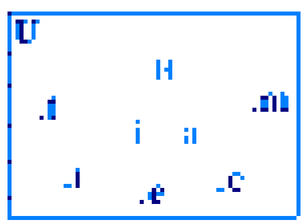
b) Sean U = { letras de la palabra aritmética} y B = { vocales de la palabra vida }

Determinado por extensión tenemos

U = { a, r, i, t, m, e, c } B = { i, a }

Su complemento de B es: B' = { r, t, m, e, c }

En forma gráfica:



2.4. Inducción matemática

Existen muchos dominios que contienen un número ilimitado de elementos. Por ejemplo, el dominio de los números naturales, el dominio de las expresiones lógicas y el dominio que consta de todos los programas escritos en Pascal son todos ellos infinitos.

Aun cuando en los dominios finitos es posible describir uno por uno todos sus elementos y se pueden enumerar sus propiedades, esto resulta imposible en los dominios infinitos.

Para generar un número infinito de elementos, uno tiene que emplear un número finito de reglas, que se podrán emplear repetidamente, o bien, para emplear un término técnico, de forma recursiva.

Otra dificultad que surge al tratar con dominios infinitos es la consistente en demostrar que todos los elementos del dominio tienen una cierta propiedad. En este aspecto, hay un método que ha demostrado sé muy versátil, y este método es la Inducción Matemática.

El primer dominio que hay que definir es el de los números naturales. En este dominio, uno comienza por el número 0, y todos los números posteriores se generan mediante una función denominada $sucesor$.

Para cada número natural n, existe un sucesor $s(n)$.; el cual no es más que $n+1$. Para demostrar que todos los elementos de un dominio tienen una cierta propiedad P, se demuestra que el 0 tiene la propiedad P, y después se demuestra que si n tiene la propiedad P entonces $s(n)$ también tiene la propiedad P.

Una vez hecho esto, el principio de Inducción Matemática nos permite concluir que P es cierto para todos los números naturales.

Aun cuando la Inducción Matemática suele estar relacionada con los números naturales, es aplicable en muchos dominios que se definen recursivamente.

En el dominio de los número naturales se pueden definir muchas funciones esenciales, tales como la suma y la multiplicación, y se hará esto mediante funciones recursivas.

Posteriormente se aplica la inducción para demostrar que estas funciones tienen ciertas propiedades.

El término de Inducción Matemática lleva en algunas ocasiones un poco a la confusión. En Filosofía, uno distingue entre deducción e inducción.

La Deducción implica derivar conclusiones basadas en argumentos lógicos y en este sentido todos los argumentos matemáticos, incluyendo la Inducción Matemática, son argumentos deductivos.

La Inducción, por otra parte implica la inferencia de reglas generales a partir de observaciones particulares, donde por supuesto el número de observaciones disponibles siempre es limitado, lo cual significa que siempre puede haber casos que violen las reglas a las que se llegue por argumentos inductivos.

La inducción matemática, tal como la inducción en un sentido filosófico, trata de las generalizaciones. Sin embargo, aun cuando la inducción normal admite excepciones para las reglas derivadas, pero la inducción matemática no las admite.

2.4.1. Principio de inducción matemática[1]

Supóngase que se tiene una proposición S(n) para cada numero entero positivo n, la cual es verdadera o falsa. Consideremos que

[1] Jonhsonbaugh, Richard, Matemáticas Discretas, Iberoamericana,1988, pag. 506.

S(1) es verdadera; a esta condición se le llama paso básico,
Si S(i) es verdadera para todo $i<n+1$, entonces S(n+1) es verdadera, y esta condición el paso inductivo
De tal manera que se cumplen los pasos anteriores para S(n) es verdadera para todo entero positivo n, es decir que la prueba por inducción matemática es satisfactoria.

2.5. Subíndices e índices[2]

La notación con subíndices es muy utilizada por su versatilidad, sobre todo cuando se utilizan una gran cantidad de elementos. Un ejemplo de ello es la utilización de polinomios, como por ejemplo:
$P(x)=a_nX^n+ a_{n-1}X^{n-1}+.................+ a_1X^1$
Donde $a_n\neq 0$
Los subíndices también son muy utilizados en conjuntos por ejemplo el conjunto $A_{n,}$ o también una sucesión de elementos $\{ x_1, x_{2,..........,} x_n\}$

2.6. Parejas ordenadas, notación matricial

2.6.1. Parejas ordenadas

En la teoría de conjuntos, se explico que un conjunto es una colección de elementos no-ordenada de elementos; es decir, un conjunto esta determinado por sus elementos no por el orden que estos tengan. Sin embargo en las matemáticas algunas ocasiones se tiene la necesidad de utilizar elementos ordenados, de aquí surge el concepto de par ordenado de elementos, el cual se denota por (x,y), el par ordenado anterior se considera diferente del par ordenado (y,x), considerando que $x\neq y$.
Una característica de los pares ordenados es que al conjunto de todos los pares ordenados se le denomina producto cartesiano.

2.6.2. Notación matricial

En muchas ocasiones es necesario es necesario expresar a los elementos enumerándolos de tal manera que cada uno de ellos tenga una posición especifica como por ejemplo una matriz de insumo producto o un sistema de ecuaciones lineales, para lo cual podemos utilizar una notación muy especifica la cual la conocemos como notación matricial, para ejemplificar esta notación lo haremos con un ejemplo, con el cual expresaremos a un sistema de ecuaciones líneas con matrices.

$A_{11}X_1+ A_{12}X_2 +A_{13}X_3 =B_1$
$A_{21}X_1+ A_{22}X_2 +A_{23}X_3 =B_2$
$A_{31}X_1+ A_{32}X_2 +A_{32}X_3 =B_3$
En el sistema de ecuaciones lineales de tres ecuaciones con tres incógnitas, se expresaran los coeficientes en una matriz, de la siguiente manera:

[2] Ross, Kenneth; Wright, Charles, Matemáticas Discretas, Prentice Hall, 2ª. Ed., 1990, pag. 673.

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

La matriz anterior representa al sistema de ecuaciones en cuanto a sus coeficientes exclusivamente, esa matriz se dice que es de 3X3, donde el primer 3 representa él numero de renglones de la matriz y el segundo él numero de columnas de la misma matriz.

2.7. Demostraciones formales

En matemáticas es normal que se requiera validar la validez de algún concepto matemático y por lo tanto se tiene la necesidad de verificar la veracidad de dicho concepto con una demostración formal.
Se llama demostración válida a la demostración formal con una sucesión válida de proposiciones, sin importar lo que se concluya. Si una o más de las proposiciones no es válida, sin importar lo que concluya, entonces el argumento es una falacia.

2.8. Métodos de demostración

Dentro de los métodos de demostración podemos mencionar, a los métodos de demostración directa y a los métodos de demostración indirecta.

2.8.1. Método de demostración directa

La demostración directa se basa en una serie de Hipótesis las cuales nos llevan a una conclusión es decir que dichas hipótesis implican razonar enforma muy concluyente.
Por ejemplo podríamos demostrar que los números naturales pertenecen a los números reales, que pi es un numero irracional, y para ello utilizar la demostración directa.

2.8.2. Método de demostración indirecta

El método de demostración indirecta es utilizado para demostración cuando se parte de una proposición contra positiva, es decir que se inicia proponiendo algo contrario, de lo que se desea probar.
Como por ejemplo se desea probar que todos los números 2n, con n en los números naturales es par, se parte diciendo con la demostración indirecta que 2n no es par.

3. Relaciones y funciones

3.1. Objetivo particular

El objetivo del presente tema es el de mostrar la utilización de funciones y relaciones para la construcciones de algoritmos recursivos.

3.2. Funciones

Definición moderna de función: Se dice que y es función de x cuando cada valor de la variable x corresponde a uno o varios valores determinados de la variable y.
La notación para expresar que y es función de x es $y=f\,(x.)$.

Cuando el valor de una variable y depende solamente del valor de otra variable x tiene una función de una sola variable independiente.
Cuando el valor de una variable y depende de los valores de dos o más valores tienen una función de varias variables independientes.

3.2.1. LEY DE DEPENDENCIA.

Siempre que los valores de una variable y depende de los valores de otra variable x, y es función de x; la palabra función indica dependencia. Pero no-basta con saber que y depende de x, interesa mucho saber como depende y de x, de que modo varia y cuando varía x, la relación que liga a las variables, que es lo que se llama ley de dependencia entre las variables.

3.2.2. CONSIDERACIONES DE DOMINIO Y RANGO.

El dominio de una función se define como el conjunto de todos los posibles valores de entrada. Nos concentraremos en las funciones de valores reales, por lo cual el dominio se compone de todos los valores reales de la variable independiente para los cuales al variable independiente se define y es real.

3.2.3. TIPO DE FUNCIONES.

Las funciones pueden clasificarse atendiendo a sus características estructurales. A continuación se explican algunas de las funciones más comunes.

3.2.3.1. FUNCIONES CONSTANTES: Una función constante tiene la forma general

$$y=f(x)=a_0$$

Donde a_0 es real. Por ejemplo, la función

$$y=f(x)=20$$

Es una función constante. Cualquiera que sea el valor de x, el rango tiene un solo valor de 20. Es decir

$$f(-10)=20$$
$$f(1.000)=20$$
$$f(a+b)=20$$

3.2.3.2. FUNCIONES LINEALES: Una función lineal tiene la forma general

$$y=f(x)=a_1x+a_0$$

Donde a_1 y a_0 son reales. La función

$$y=f(x)=-2x+15$$

Es una función lineal con $a_1=-2$ y $a_0=15$

3.2.3.3. FUNCIONES CUADRÁTICAS: Una función cuadrática tiene la forma general

$$y=a_2x^2+a_1x+a_0$$

Donde a_2, a_1 y a_0 son reales y a_2 es diferente de 0 .La función

$$y=f(x)=3x2-20x+100$$

Es una función cuadrática y en ella $a_2=3$, $a_1=-20$ y $a_0=100$

La función

$$y=f(x)=-\frac{X2}{2}$$

Es una función cuadrática con $a_0=a_1=0$ y $a_2=-1/2$.

Ejemplo: La función de demanda es una relación matemática que expresa la forma en que la cantidad de la demanda de un producto varía según el precio en que se venda. La función de demanda de un producto determinado es:

$$q_d=f(p)$$

o bien $$q_d=p^2-70p+1225$$

Donde q^d es el número de unidades de demandas y p es el precio en dólares. Nótese que la función de demanda tiene forma cuadrática donde a2 = 1, a2 =-70 y a0 = 1225.De acuerdo con esta función, la cantidad de la demanda a un precio de $10 se espera que sea igual a

$$f(10)=(10)^2-70(10)+1225$$
$$=100-700+1225$$
$$=625 \text{ unidades}$$

Y a un precio de $30 $$f(30)=f(30)^2-70(30)+1225$$
$$=900-2100+1225=25 \text{ unidades}$$

3.3. Funciones invertibles

De una manera burda se puede decir que si una función tiene inversa o es invertible entonces la función inversa anula la acción de la función.

Para que una función sea inversa debe cumplir con la condición de que sea uno a uno.

Lo anterior quiere nos indica, que si es uno a uno entonces debe cumplir para cada elemento del dominio le corresponde un elemento del contradominio y viceversa, es decir para cada elemento del contradominio le corresponde un elemento del dominio.

3.4. Sucesiones y notación o_grande

3.4.1. Sucesiones

Una familia importante de funciones es por ejemplo la que consta de las funciones que tienen como dominio el conjunto de los números naturales, estas funciones se llaman sucesiones.

3.4.2. Notación o_grande

La notación o_grande es utilizada para describir estimación que representa un numero de procesos o pasos que se realizan n veces, y que se utilizan de manera constante sobre todo en el área de computación.

Para ejemplificar esto supongamos que tenemos una función f(n), para que esta función sea una o_grande necesitamos acotar a la función f(n), y de esta manera obtendríamos nuestra o_grande O(n).

3.5. Definiciones recursivas

Recursión[3] es la forma en la cual se especifica un proceso basado en su propia definición. Siendo un poco más precisos, y para evitar el aparente círculo sin fin en esta definición, las instancias *complejas* de un proceso se definen en términos de instancias más *simples*, estando las **finales** más simples definidas de forma explícita.

La recursividad[4] es una técnica importante de programación que permite que una función se llame a sí misma. Como ejemplo útil se puede presentar el cálculo de números factoriales. El factorial de 0 está definido específicamente como 1. El factorial de *n*, un entero mayor que 0, es el producto de todos los enteros del intervalo comprendido entre 1 y *n*.

La recursividad es una técnica de programación importante. Se utiliza para realizar una llamada a una función desde la misma función. Como ejemplo útil se puede presentar el cálculo de números factoriales. Él factorial de 0 es, por definición, 1. Los factoriales de números mayores se calculan mediante la multiplicación de 1 * 2 * ..., incrementando el número de 1 en 1 hasta llegar al número para el que se está calculando el factorial.

3.6. Relaciones recursivas[5]

Las sucesiones que se definen por recursión aparecen con frecuencia en matemáticas, hay muchas formas para obtener formulas explicitas de ellas, en seguida se muestra una formula para relaciones recursivas:

$S_n=as_{n-1} + bs_{n-2}$

Donde a y b son constantes, y los valores de s_0 y s_1 son especificados.

[3] (Fuente: http://es.wikipedia.org/wiki/Recursividad)
[4] http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/jscript7/html/recurse.asp

[5] Ross, Kenneth; Wright, Charles, Matemáticas Discretas, Prentice Hall, 2ª. Ed., 1990, pag. 673.

3.7. Definiciones generales de recursion

Generalizando podemos decir que un conjunto de objetos esta definido recursivamente siempre que:

Algunos elementos del conjunto se especifican explícitamente, esta es **la base** de la definición,
El resto de los elementos del conjunto se definen en términos de los elementos ya definidos a esta se le denomina la **cláusula recursiva**.

3.8. Relaciones de equivalencia[6]

Una relación para que sea de equivalencia debe cumplir las siguientes propiedades reflexiva, simétrica y transitiva.

Ejemplo 1: La relación *R* es definida sobre el conjunto de cadenas de letras del alfabeto latino, tal que *aRb* si *l*(*a*) = *l*(*b*), donde *l*(*x*) es el largo de *x*. ¾Es *R* una relación de equivalencia?
Solución: *R* es reflexiva, ya que *l*(*a*) = *l*(*a*) y por ende *aRa*.
Si *aRb*, entonces *l*(*a*) = *l*(*b*). Pero el *l*(*b*) = *l*(*a*) por las propiedades de igualdad, por tanto *bRa* y *R* es simétrica.
R es transitiva, ya que si *l*(*a*) = *l*(*b*) y *l*(*b*) = *l*(*c*), entonces *l*(*a*) = *l*(*c*). /

Ejemplo 2: *R* : R *$* R, *R* = *f*(*a; b*)*j*(*a i b*) *2* Z*g* ¾Es *R* una relación de equivalencia?
Solución: Ya que *8a* (*a i a*) = 0 y 0 es un entero, *8a aRa* y *R* es reflexiva.
Si (*a i b*) *2* Z, entonces (*b i a*) *2* Z. Por tanto, *8a8b*(*aRb) bRa*) y *R* es simétrica.
Si (*a i b*) *2* Z y (*b i c*) *2* Z, entonces (*a i b*) *i* (*b i c*) = (*a i c*) *2* Z. Por tanto, *8a8b8c*((*aRb ^ bRc*) *) aRc*) y *R* es transitiva, /

Ejemplo 3: Sea *R* : Z *$* Z, *R* = *f*(*a; b*)*ja ´ b*(*mod m*)*g* se llama relación de congruencia módulo *m*, *m 2* Z y *m >* 1 ¾Es *R* una relación de equivalencia?
Solución: *a ´ b*(*mod m*), si y sólo si *m* divide (*a i b*).
Ya que (*a i a*) = 0 se divide por *m*, entonces *a ´ a*(*mod m*). Por tanto, la congruencia módulo *m* es re_exiva.
Ahora supongamos que *a ´ b*(*mod m*). Si (*a i b*) = *km*, entonces (*b i a*) = (*ik*)*m*, lo cual significa que *b ´ a*(*mod m*) y *R* es simétrica.
Supongamos que *a ´ b*(*mod m*) y *b ´ c*(*mod m*). Entonces, (*aib*) = *km* y (*bic*) = *lm*.
Sumando las dos ecuaciones, tenemos *a i c* = (*a i b*) *i* (*b i c*) = *km* + *lm* = (*k* + *l*)*m*, lo que significa que *a ´ c*(*mod m*), y que *R* es transitiva. /
Definición: Sea *R* una relación de equivalencia sobre el conjunto *A*. El conjunto de elementos relacionados a un elemento *a 2 A* se llama clase de equivalencia de *a*. La clase de equivalencia de *a* respecto a *R* se denota como [*a*]*R*. Cuando se considera una sola relación, se puede omitir *R*: [*a*].
[*a*]*R* = *fsj*(*a; s*) *2 Rg*
Si *b 2* [*a*]*R*, entonces *b* se llama representante de la clase de equivalencia de *a*.

3.9. Relaciones generales

Dentro de las relaciones generales tenemos a las siguientes:

[6] http://www.inf.utfsm.cl/~liuba/fund/rel_equival.pdf#search='relaciones%20de%20equivalencia'

Relación binaria

Relación de equivalencia

Relación inversa

3.10.Composición de relaciones

La composición de relaciones esta dada por $R_{g\,o\,f}$ donde R_g y R_f están dadas por
$R_f = \{<x,y> \in S \text{ x } T: y=f(x)\}$
$R_g = \{<x,y> \in S \text{ x } T: y=g(x)\}$

De donde la composición de R_f y R_g esta dada por
$R_{g\,o\,f} = \{<x,z> \in S \text{ x } T: z=g(f(x))\}$

3.11.Cerradura

En algunas ocasiones quisiéramos formar nuevas relaciones a apartir de algunas que ya tenemos, entonces para poder hacer esto las la relación tiene que ser reflexiva, simétrica y transitiva.

4. Análisis de algoritmos

4.1. Objetivo particular

El objetivo del presente tema es el de analizar todas estructuras de los árboles y así por construir algoritmos eficientemente.

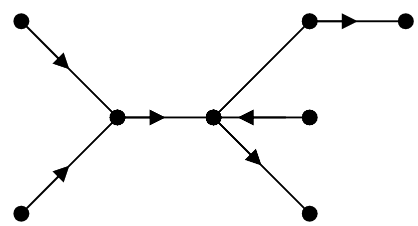
4.2. Propiedades de los árboles

Dentro de las propiedades podemos mencionar que un árbol es una gráfica aciclica conexa.
Además en particular un árbol no tiene lazos ni aristas paralelas.
Si a un árbol se le quitan algunas aristas, pero manteniendo los vértices, a este árbol se le llama árbol generador.
Estas son algunas de las propiedades importantes de los árboles.

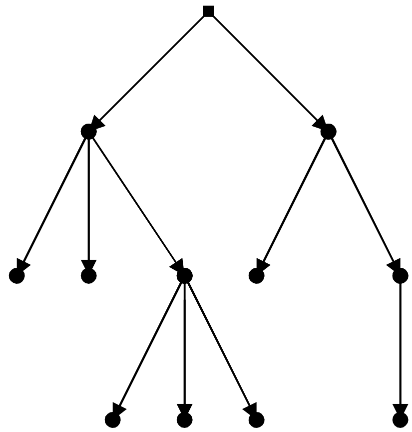
4.3. Árboles enraizados

Ahora bien, un árbol dirigido es cuando se ignoran las direcciones de sus aristas. Por ejemplo:

ÁRBOL DIR**IGIDO**



Un árbol enraizado es cuando exactamente un vértice cuyo grado de entrada sea 0 y los grados de entrada de todos los otros vértices sean 1. Por ejemplo:



2

A los árboles enraizados también se les conoce como árboles familiares.

En muchas ocasiones encontramos estructuras que pueden representarse como árboles enraizados. Por ejemplo el organigrama de un corporación como en la figura A se muestra.

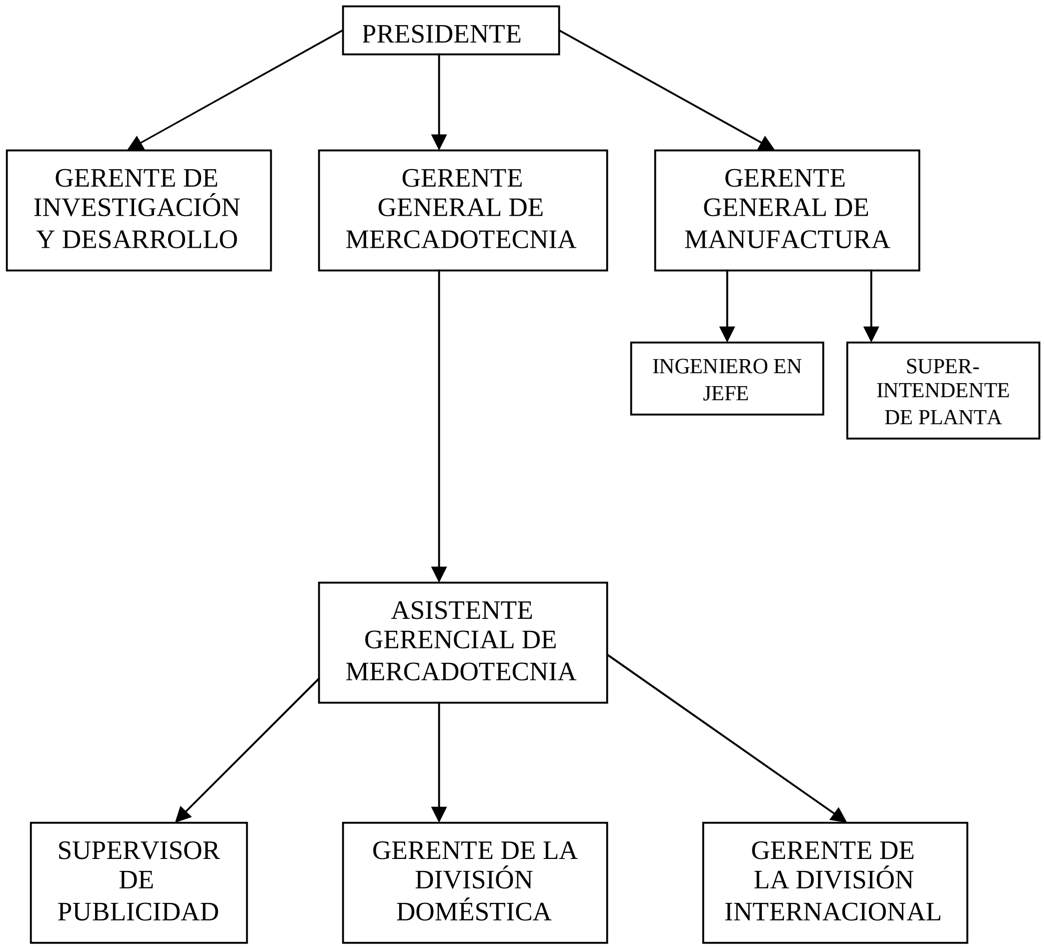
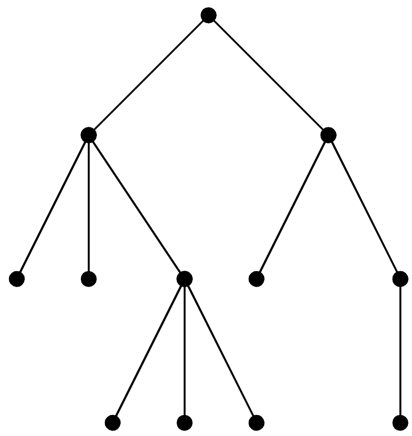


Fig. A

3

Cuando trazamos un árbol enraizado, si nos apegamos a la convención de colocar los hijos de un nodo rama bajo éste, las puntas de flecha de las aristas pueden omitirse, debido a que puede entenderse que apuntan hacia abajo. Por ejemplo:



Consideremos el árbol enraizado de la fig. 2 el cual es el árbol familiar de un hombre que tiene dos hijos, de los cuales el mayor no tiene hijos y el menor tiene tres hijos.
A pesar de que el árbol enraizado de la figura 3 es isomorfo al de la figura 2, podría ser el árbol familiar de otro hombre cuyo hijo mayor tiene tres hijos y el hijo menor no tiene hijos.

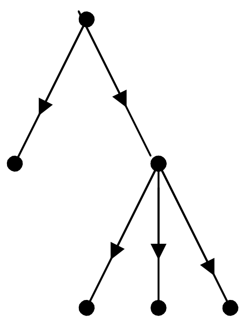


Fig. 2

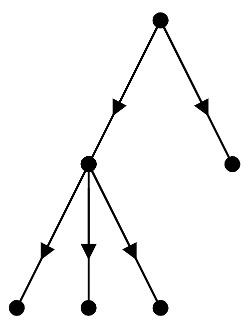


Fig. 3

4

4.4. Algoritmos de búsqueda de primera profundidad

El recorrido por profundidad sigue primero una trayectoria desde el nodo inicial hasta un nodo terminal, después otra trayectoria desde el mismo punto inicial hasta alcanzar otro final., y así sucesivamente hasta que todos los nodos hayan sido visitados.

1,2,4,8,5,7,3,6

1,3,6,7,8,2,5,4

DEFINICION

Un grafo G = (N,A,☐) consta de un conjunto no vacío N denominado conjunto de nodos (puntos , vértices) del grafo, un conjunto A de aristas del grafo y una correspondencia ☐ del conjunto de aristas A en un conjunto de pares ordenados o desordenados de N. Si una arista se corresponde con un par ordenado, entonces se dice que es una arista dirigida, es caso contrario, se denomina arista no dirigida.

Obsérvese que la definición de grafo implica que a toda arista del grafo G se le puede asociar una pareja ordenada o desordenada de nodos del grafo. Si una arista e ☐ A está asociada de esta forma con un par ordenado (u,v) o con un par desordenado { u,v } , en donde u,v ☐ N, entonces se dice que la arista e conecta o une los nodos u y v.

Los pares de nodos que estén conectados por una arista dentro de un grafo se denominan nodos adyacentes. Se supondrá en todo momento que tanto el conjunto A como el conjunto N de un grafo son finitos. Con frecuencia será necesario escribir los grafos en la forma (N,A) o bien simplemente como G. En el primer caso, cada arista se representa directamente como el par con el cual se corresponde, lo cual obvia la necesidad de especificar ☐ si ☐ es una correspondencia uno a uno.

EJEMPLO

Una de las aplicaciones más frecuentes de los grafos es la que se tiene cuando planea uno sus vacaciones. Si viajamos por carretera , se utilizan los mapas de carreteras que representan la red viaria disponible para el viaje. Un mapa de carreteras es un grafo en el cual nos nodos son los pueblos y ciudades de alguna comarca, y las aristas representan las carreteras que unen estos pueblos y ciudades.

La figura 4.2.1 muestra un mapa de las autopistas más importantes que unen entre si las ciudades principales del Oeste de Canadá. Todas las aristas o carreteras del grafo pueden ser recorridas en ambas direcciones. El número (o peso) que se asocia a cada arista denota la distancia en kilómetros existente entre las dos ciudades que une esa arista (por ejemplo, Winnipeg y Victoria pasando por Edmonton). Un viajero, como consecuencia de las limitaciones de tiempo, podría estar interesado también en la distancia mínima entre dos ciudades dadas.

Una herramienta de uso frecuente por parte de quienes hacen diseños urbanísticos y de transportes es la simulación por computadora de sistemas de tráfico. Los sistemas que se modelan van desde las redes de tráfico nacionales, a las calles de una ciudad, pasando por ciertas zonas urbanas y llegando, incluso , al tráfico existente en un cierto puente o cruce de carreteras.

Los modelos se utilizan para poner de manifiesto puntos negros actuales o futuros y para sugerir y probar cambios propuestos o nuevos sistemas.

En una ciudad, el sistema de calles puede modelar como un grafo en el cual los cruces se representan como nodos y los segmentos de calle existentes entre cruces son las aristas. Las calles de doble sentido se representan como grafos no dirigidos (esto es, bordes sin flechas) mientras que las calles de dirección única son aristas dirigidas .

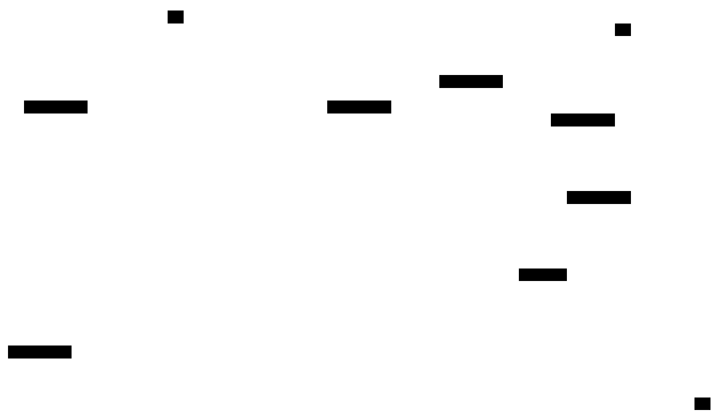
La figura 4.2.2(a) muestra una parte del plano de una ciudad, en el cual las flechas denotan calles de dirección única. Los bomberos y la policía están interesados en los caminos más cortos desde una comisaría de policía o parque de bomberos hasta el lugar en que se produce una llamada 091 pidiendo ayuda.

La figura 4.2.2(b) es una abstracción de un sistema de calles de una ciudad, en el cual las aristas estarían rotuladas con nombres de calles, densidades de tráfico y cosas parecidas.

EJEMPLO

Una aplicación más reciente de los grafos es el modelado de redes de computadoras. Típicamente, una red de computadoras consta de toda una gama de elementos, tales como computadoras y líneas de comunicación.

En la representación de una red de computadoras mediante un grafo, cada nodo es un dispositivo , tal como una computadora o una terminal, y cada arista o enlace denota un medio de comunicación, tal como una línea telefónica o un cable de comunicación. Muchas compañías y universidades poseen una o más redes de área local, que típicamente abarcan menos de 1 kilómetro cuadrado. Sin embargo , existen muchas redes de larga distancia o redes remotas cuyos vértices, desde un punto de vista geográfico , pueden abarcar uno o más países. Los grafos son importantes para modelar estas redes con respecto a su fiabilidad y a su eficiencia.



En la figura 4.2.3 se ha representado un grafo de una red de computadoras. La parte subred representa la parte de comunicaciones de la red.

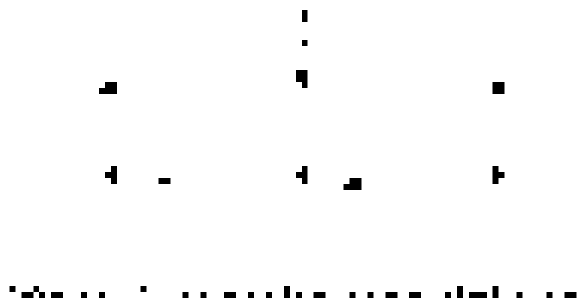
Los demás dispositivos que se encuentran alrededor de la subred de comunicaciones se pueden considerar dispositivos externos. Aun cuando nos hemos tomado la libertad de representar mediante iconos los terminales y las Pc., hay que dejar claro que en un grafo abstracto estos componentes serían nodos o vértices.

La disposición de vértices de subred pueden estar muy alejados y dispuestos de formas arbitrarias.

EJEMPLO

Hay muchos programas que constan de módulos que se invocan unos a otros. Los grafos de llamadas representan módulos mediante nodos. Una línea dirigida que va del nodo x al nodo y indica que x invoca a y, en el grafo de llamadas de la figura 4.2.4 , por ejemplo el modulo A invoca a los módulos B,C y D.

Los módulos B y C invocan al módulo E. Cuando uno de los módulos invoca a otro, tiene que haber una comunicación entre esos módulos a través de una interfaz.



NOTA: Los ejemplos anteriores son ejemplos de grafos sencillos.

4.5. Notación polaca

La notación polaca puede ser utilizada para escribir expresiones que involucren objetos de algunos sistemas y algunas operaciones en lo objetos.
Las operaciones son normalmente, aunque no siempre, operaciones binarias.

4.6. Árboles pesados

Un grafo es pesado cuando sus aristas contienen datos (etiquetas). Una etiqueta puede ser un nombre, costo ó un valor de cualquier tipo de dato. También a este grafo se le denomina ***red de actividades***, y el número asociado al arco se le denomina ***factor de peso***.
Tanto a las aristas como a los vértices les puede ser asociada información .A esta información se le llama etiqueta. Si la etiqueta que se asocia es un número se le llama peso, costo o longitud. Un grafo cuyas aristas o vértices tienen pesos asociados recibe el nombre de **grafo etiquetado o ponderado.**

5. Algoritmos en grafos

5.1. Objetivo particular

El objetivo del presente tema es el de crear algoritmos para grafos y a su vez identificar sus posibles modificaciones para producir algoritmos alternativos.

5.2. Algoritmos para graficas

El estudio de las gráficas nos ha llevado a un numero de innumerable de pregunta para ello es necesario contar con algoritmos que den respuesta a estas preguntas.

Entre estos algoritmos tenemos los siguientes:

5.2.1. Algoritmo de Dijskstra

Este algoritmo encuentra únicamente pesos mínimos desde un vértice seleccionado a los demás vértices en la gráfica.

5.2.2. Algoritmo de Warshall

Este algoritmo produce la matriz de peso mínimo.

5.2.3. Algoritmo de Max-Weight

Este algoritmo encuentra únicamente pesos máximos desde un vértice seleccionado a los demás vértices en la gráfica.

5.3. Modificaciones

Las versiones de los algoritmos Dijkstra y de Warshall producen pesos mínimos y máximos.
Por ello con algunas modificaciones estos algoritmos Dijkstra y de Warshall pueden modificarse para que dichos algoritmos creen los caminos correspondientes, es decir los camino mínimo y el camino máximos.