

Actividad 1

Estructura Secuencial

Para cada uno de los siguientes ejercicios, escriba el algoritmo utilizando diagrama de flujo y pseudocódigo. No olvide definir entradas y salidas claramente. Finalmente, proporcione el numero de operaciones que el algoritmo llevara a cabo.

1. Escribir un algoritmo para convertir una temperatura dada en grados Fahrenheit a grados Celsius.
2. Escribir un algoritmo que a partir de las datos de la base y la altura de un rectángulo, calcule su área.
3. Escribir un algoritmo que dados los 3 coeficientes reales a , b , c de un polinomio de segundo grado, determine sus raíces reales.

Actividad 2

Estructura condicional

Para cada uno de los siguientes ejercicios, escriba el algoritmo utilizando diagrama de flujo y pseudocódigo. No olvide definir entradas y salidas claramente. Finalmente, proporcione el numero de operaciones que el algoritmo llevara a cabo.

1. Escribir un algoritmo que dados 3 números enteros indique cuál posee el valor máximo.
2. Escribir un algoritmo que dados 3 números reales determine cuál es el mínimo valor del producto de dos de estos números.
3. Escribir un algoritmo que dado un año indique si es bisiesto o no lo es, teniendo en cuenta que son bisiestos todos los múltiplos de 4 excepto los múltiplos de 100 (p.e. el 1900 no fue bisiesto), aunque los múltiplos de 400 sí que son bisiestos (por eso lo ha sido el año 2000).

Actividad 3

Estructura iterativa

Para cada uno de los siguientes ejercicios, escriba el algoritmo utilizando diagrama de flujo y pseudocódigo. No olvide definir entradas y salidas claramente. Finalmente, proporcione el número de operaciones que el algoritmo llevara a cabo.

1. Escribir un algoritmo que calcule el máximo común divisor de dos números naturales positivos según el algoritmo de Euclides, que consiste en ir restando el número menor al mayor hasta obtener dos números iguales, momento en el cual tenemos el máximo común divisor.
2. Escribir un algoritmo que muestre los cubos de los primeros números naturales tales que el cubo sea menor que n .
3. Escribir un algoritmo que dado un número natural, calcule su factorial.
4. Escribir un algoritmo que trate de adivinar el número que está pensando una persona (entre 0 y 99) en un máximo de 10 intentos.
El algoritmo podrá proponer un número y recibir indicación si el número pensado es mas algo o mas bajo.

Actividad 4

Ciclos anidados

Para cada uno de los siguientes ejercicios, escriba el algoritmo utilizando diagrama de flujo y pseudocódigo. No olvide definir entradas y salidas claramente. Finalmente, proporcione el número de operaciones que el algoritmo llevara a cabo.

1. Escribir un algoritmo que muestre todas las fichas de un dominó, en la forma siguiente:

0-0

1-0 1-1

2-0 2-1 2-2

3-0 3-1 3-2 3-3

4-0 4-1 4-2 4-3 4-4

5-0 5-1 5-2 5-3 5-4 5-5

6-0 6-1 6-2 6-3 6-4 6-5 6-6

2. Escriba un algoritmo que dado un conjunto de puntos en el plano $(X_1, Y_1), \dots, (X_n, Y_n)$, encuentre la pareja de puntos más cercana entre sí.

3. Se tiene un archivo de n registros y cada registro contiene m datos que deben actualizarse periódicamente. Diseñe un algoritmo que actualice tal archivo. Dicho algoritmo recibe como entrada el número de registros en el archivo y una lista que contiene los números de registro que se han de actualizar. Si un número de registro está en la lista, todos sus datos se actualizan.

4. Escriba un algoritmo que calcule el producto de dos matrices de $N \times N$.

Actividad 5

Complejidad Algorítmica

1. El mínimo común múltiplo (MCM) de dos enteros positivos m y n , es el entero más pequeño que es un múltiplo de ambos números. Diseñe un algoritmo iterativo para encontrar el MCM de dos enteros. Determine el orden de su complejidad temporal.
2. Escriba la función que define el tiempo de ejecución $T(n)$ para el mejor y peor caso del siguiente algoritmo. Escriba el procedimiento para llegar a la respuesta.

//Entrada A[0 ... n - 1]

desde i ← 0 hasta n - 2 hacer

desde j ← i + 1 hasta n - 1 hacer

si A[i] = A[j] regresar falso

regresar verdadero

3. Escriba la función que define el tiempo de ejecución $T(n)$ para el mejor y peor caso del siguiente algoritmo. Escriba el procedimiento para llegar a la respuesta.

//Entrada T[1 ... n]

desde i ← 1 hasta n - 1 hacer

minj ← i

minx ← T[i]

desde j ← i + 1 hasta n hacer

si T[j] < minx

minj ← j

minx ← T[j]

T[minj] ← T[i]

T[i] ← minx

4. Dos algoritmos requieren n^2 días y n^3 segundos respectivamente para resolver casos de tamaño n . ¿Cuál es el tamaño del caso más pequeño en el cual el primer algoritmo es más rápido que el segundo? ¿Aproximadamente, cuánto tiempo tarda en resolver este caso?

¿Cuáles de las siguientes afirmaciones son verdaderas? Demuestre sus respuestas:

a. $n^2 \in O(n^3)$

b. $n^2 \in \Omega(n^3)$

c. $2^n \in \theta(2^{n+1})$

d. $n! \in \theta((n+1)!)$

5. Describa que hacen los siguientes algoritmos y determine su O-grande, Omega, Theta

a)

función T (var A:matriz)

var i,j,filas: entero fvar

filas=numfilas (A);

para i=1 hasta filas-1 hacer

para j=i+1 hasta filas hacer

b=A[j,i];

A[j,i]=A[i,j];

A[i,j]=b;

fpara

fpara

fin

b)

for(i=1; i<=n; i++);

for(j=1; j<=i;j++);

sum ++;

for(k=1; k<=n; k++);

A[k] = k - 1;

c)

for (int i= 0; i < N; i++) {

c= i;

while (c > 1) {

algo_de_O(1)

c= c/2;

}}

Actividad 6

Algoritmos de Fuerza Bruta

1. Tres equipos de investigación tratan de resolver un mismo problema de forma independiente. Las probabilidades de fracasar son 0.4, 0.6 y 0.8. Se desea minimizar la probabilidad de fracaso y se dispone de dos científicos más para reforzar los equipos. Para determinar a que equipo asignarlos se elabora la tabla siguiente, que da la probabilidad de fracaso de cada equipo cuando es reforzado con 0, 1 y 2 científicos.

Equipo			
Refuerzos	1	2	3
0	0.4	0.6	0.8
1	0.2	0.4	0.5
2	0.15	0.2	0.3

2. Escribe un algoritmo que, utilizando únicamente operaciones básicas (+, -, *, /) sea capaz de obtener el valor aproximado a cuatro espacios decimales de la raíz cuadrada de cualquier número. Prueba con las siguientes operaciones $\sqrt{7}$, $\sqrt{26}$ y $\sqrt{50}$

Actividad 7

Algoritmos voraces

1. Suponga que tiene una lista de objetos con diferentes pesos en kilogramos y valores en pesos [peso, valor]. Usted tiene una mochila que puede cargar un peso máximo de X kilogramos. Diseñar un algoritmo voraz que elija los objetos de la lista que maximicen el valor total de los objetos que se llevarán en la mochila [peso, valor]:

([87, 33], [45, 12], [20, 16], [54, 23], [45, 42], [26, 12], [30, 13], [54, 21], [76, 47], [12, 22], [64, 44], [23, 11])

y una capacidad de 280K.

2. Dado un sistema monetario S de longitud K y una cantidad de cambio C , devolver una solución (si existe) que nos indique el número de monedas de S equivalente a C , es decir, que nos muestre el cambio para C a partir de monedas de S

Actividad 8

Algoritmos de Divide y vencerás

1. Diseñe un algoritmo para encontrar el mínimo y máximo de un arreglo utilizando divide y vencerás. Haga el análisis del algoritmo.
2. Cuenta la leyenda que en un templo hay tres varillas de diamante con 64 discos ordenados de mayor a menor en la primera, y que unos monjes están encargados de mover los discos a otra varilla. Las reglas para mover los discos son que solamente se pueden mover de uno en uno y que nunca puede haber un disco grande sobre uno más pequeño. Según la leyenda, el mundo se acabará el día que los monjes terminen de mover los discos. Diseña un algoritmo recursivo para resolver el problema. Determina su eficiencia.

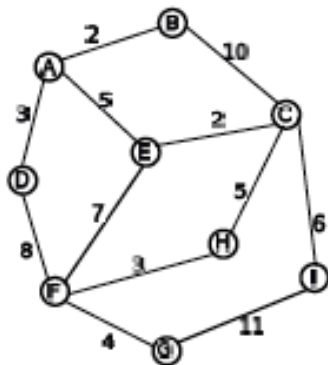
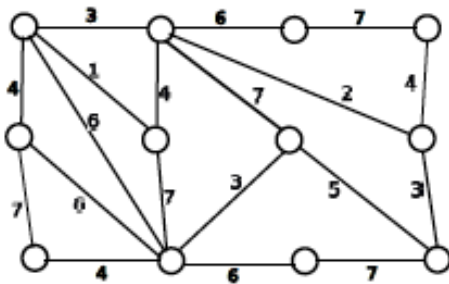
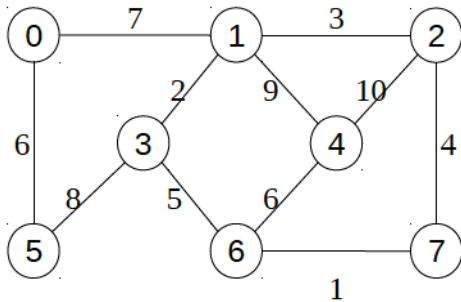
Actividad 9

Backtracking

1. Se dispone de un tablero rectangular, por ejemplo el tablero de ajedrez, y de un caballo, que se mueve según las reglas de este juego. El objetivo es encontrar una manera de recorrer todo el tablero partiendo de una casilla determinada, de tal forma que el caballo pase una sola vez por cada casilla. Diseñe un algoritmo para solucionar este problema utilizando backtracking

Actividad 10: Algoritmoa de Kruskal y Prim

1. Obtenga el árbol de expansión mínima de los siguientes grafos usando a) el algoritmo de Kruskal y b) el algoritmo de Prim. Dibuje la progresión de nodos y arcos seleccionados



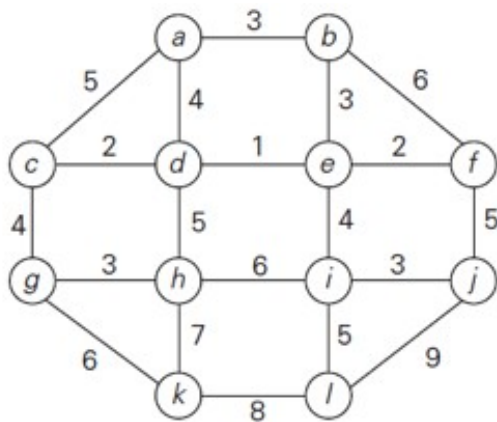
Actividad 11

Algoritmo de Dijkstra

1. Para el siguiente grafo, utilice el algoritmo de Dijkstra para encontrar el camino mas corto de:

- a) Nodo (a) a nodo (i)
- b) Nodo (h) a nodo (b)
- c) Nodo (k) a nodo (b)
- d) Nodo (g) a nodo (f)

Dibuje los pasos para las rutas de cada uno de los incisos



2.- Considérese la información registrada en una agencia de viajes referente a los vuelos existentes entre todos los aeropuertos del mundo. Supongamos que toda conexión aérea es de ida y vuelta. Queremos resolver el siguiente problema: Dados los aeropuertos A y B ¿Cuál es el mínimo número de transbordos en un viaje aéreo de A a B?

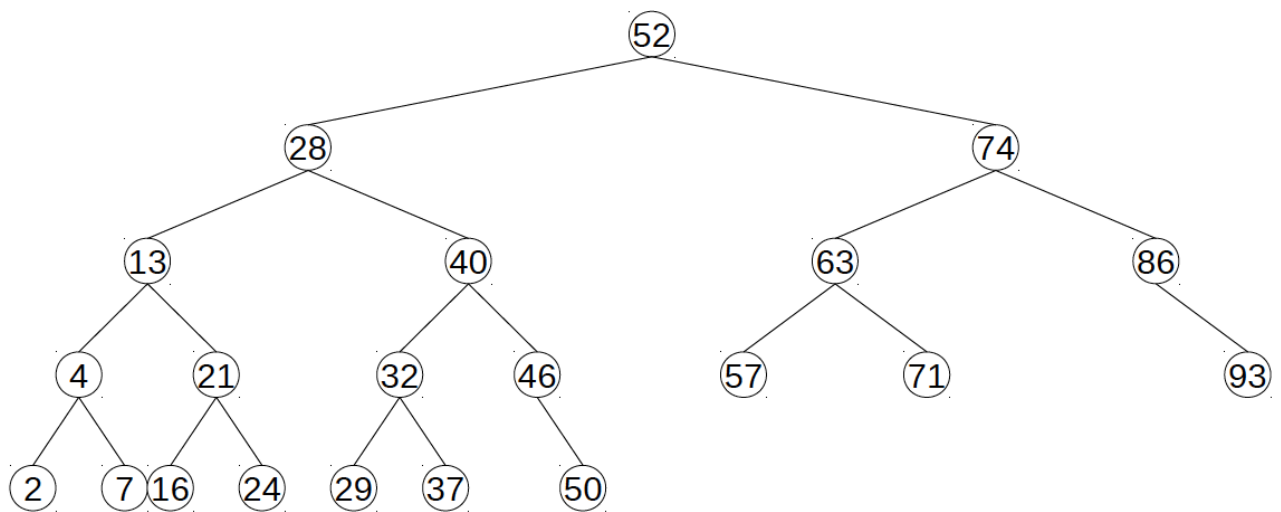
- (a) De los algoritmos de recorrido de grafos conocidos, ¿cuál elegirías para resolver este problema y por qué?. Indica cuál sería la modificación pertinente del algoritmo elegido.
- (b) ¿Cuál es la complejidad temporal de cada uno de los algoritmos sobre grafos que conoces que resuelvan este problema?
- (c) En conclusión, ¿cuál de ellos elegirías y por qué?.

Actividad 12

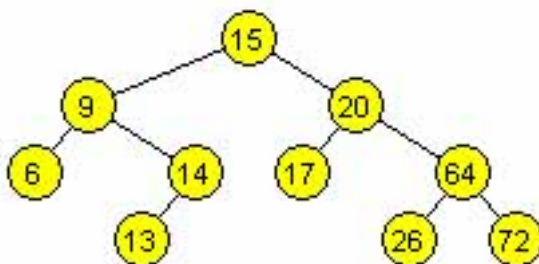
Árboles binarios de búsqueda

1. Dibuja el árbol binario de búsqueda resultante después de haber realizado las siguientes operaciones sobre el mismo:

- Insertar 54
- Eliminar 40
- Eliminar 74
- Eliminar 86
- Eliminar 63



2. Dado el siguiente árbol



escribir el recorrido en:

- inorden
- preorden
- postorden

Actividad 13

Algoritmos recursivos

1. Escribe dos algoritmo que permitan encontrar el factorial de un número. El primero de manera iterativa y el segundo de manera recursiva. Analizar los algoritmos en cuanto su complejidad en numero de operaciones y uso memoria.
2. Escribir una algoritmo recursivo para calcular la potencia de un número real elevado a un exponente entero.
3. Escribir un algoritmo recursivo para calcular el término i -ésimo de la serie de Fibonacci.