**ALGORITMOS**

**DEFINICIÓN:** es un conjunto de instrucciones, ordenadas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad. En la vida cotidiana, se emplean algoritmos frecuentemente para resolver problemas. Algunos ejemplos son los manuales de usuario, que muestran algoritmos para usar un aparato, o las instrucciones que recibe un trabajador por parte de su patrón y también Algunos ejemplos en [matemática](http://es.wikipedia.org/wiki/Matem%C3%A1tica) como el [algoritmo de multiplicación,](http://es.wikipedia.org/wiki/Algoritmo_de_multiplicaci%C3%B3n) el de la [división](http://es.wikipedia.org/wiki/Divisi%C3%B3n_%28matem%C3%A1ticas%29) y entre otros.

**PSEUDOCÓDIGO:** es una descripción de [alto nivel](http://es.wikipedia.org/wiki/Lenguaje_de_alto_nivel) compacta e informal del principio operativo de un [programa informático](http://es.wikipedia.org/wiki/Programa_inform%C3%A1tico). Utiliza convenciones estructurales de un [lenguaje de programación](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) pero está diseñado para la lectura humana en lugar de la lectura mediante máquina, y con independencia de cualquier otro lenguaje de programación. Normalmente, el pseudocódigo omite detalles que no son esenciales para la comprensión humana del algoritmo, tales como declaraciones de variables, código específico del sistema y algunas [subrutinas](http://es.wikipedia.org/wiki/Subrutina). El lenguaje de programación se complementa, donde sea conveniente, o con notación matemática compacta. Se utiliza pseudocódigo pues este es más fácil de entender para las personas que el código del lenguaje de programación convencional, ya que es una descripción eficiente y con un entorno independiente de los principios fundamentales de un algoritmo.

**CLASES DE ALGORITMOS**

**Algoritmos de Fuerza Bruta:** Resuelven el problema con la estrategia más obvia de solución, que no siempre es la mejor según el número de operaciones que se requiere.

**Ejemplo:**

El algoritmo encuentra todas las ocurrencias del patrón en el texto.
-El **patrón** se denota por x = x [0,..., m-1]; su longitud es igual a m.

-El **texto** se denota por y = y [0,..., n-1]; su longitud es igual a n.

-Ambas secuencias son estructuras sobre un sistema finito de caracteres llamado alfabeto denotado por S, con tamaño igual a s.

**x = GCAGAGAG**
**y = GCATCGCAGAGAGTATACAGTACG**

**ALGORITMOS DE BUSQUEDA:** es aquel que está diseñado para localizar un elemento con ciertas propiedades dentro de una estructura de datos; por ejemplo, ubicar el registro correspondiente a cierta persona en una base de datos, o el mejor movimiento en una partida de ajedrez.

**Ejemplo:** O( **log2 n** ). Recordemos que el arreglo debe estar ordenado para poder aplicar este algoritmo.

**ALGORITMO DE EXPLORACION DE GRAFO**:Los algoritmos de búsqueda en grafos nacen por la necesidad de crear un mecanismo de navegación autónoma, bien sea de robots, coches, o personajes en un videojuego.

**Ejemplo:** 2=3,4=4... 5

**ALGOITMOS HEURISTICOS:**el propósito de estos algoritmos no es necesariamente encontrar una solución final al problema, sino encontrar una solución aproximada cuando el tiempo o los recursos necesarios para encontrar la solución perfecta son excesivos.

**Ejemplo:**

Un problema clásico que usa heurísticas es el [puzzle-n](http://es.wikipedia.org/w/index.php?title=Puzzle-n&action=edit&redlink=1). Contar el número de casillas mal colocadas y encontrar la suma de la [distancia Manhattan](http://es.wikipedia.org/wiki/Distancia_Manhattan) entre cada bloque y su posición al objetivo son heurísticas usadas a menudo para este problema. Se realiza a partir de la categoría gramatical.

**ALGORITMOS VORACES:** seleccionan la opción de solución (solución local) que tenga un costo menor en la etapa de solución en la que se encuentran, sin considerar si esa opción es parte de una solución óptima para el problema completo (solución global).

**Ejemplo:**

Encontrar la secuencia óptima para procesar un conjunto de tareas por un computador, hallar el camino mínimo de un grafo, etc. Habitualmente, los elementos que intervienen son:

* un conjunto o lista de ***candidatos*** (tareas a procesar, vértices del grafo, etc.);
* un conjunto de ***decisiones*** ya tomadas (candidatos ya escogidos);
* una ***función*** que determina si un conjunto de candidatos es una ***solución*** al problema (aunque no tiene por qué ser la óptima);
* una ***función*** que determina si un conjunto es ***completable***, es decir, si añadiendo a este conjunto nuevos candidatos es posible alcanzar una solución al problema, suponiendo que esta exista;
* una ***función*** de selección que escoge el candidato aún no seleccionado que es más ***prometedor***;
* una ***función objetivo*** que da el valor/coste de una solución (tiempo total del proceso, la longitud del camino, etc.) y que es la que se pretende maximizar o minimizar;

**QUE SON VARIABLES**

Son zonas de memoria cuyo contenido cambia durante la fase de procesamiento de información.

**Tipos de variables:**

Variables Numéricas (Enteras y Reales)

Variables Alfanuméricas

a) Caracteres alfabéticos

b) Dígitos

 c) Caracteres especiales

Variables Lógicas (Boolenas) Son las que únicamente pueden contener dos valores, los cuales son:

Valor Verdadero ó 1 Valor Falso ó 0

Estos tipos de variables pueden tomar únicamente valores del mismo tipo, es decir si la variable es entera solamente puede almacenar datos enteros.

**Ejemplo:** Una variable declarada como cadena solo puede tomar valores correspondientes a ese tipo: Nombre = "Manuel López"

**QUE SON CONSONANTES:**

Una constantees un dato numérico o alfanumérico que no cambia durante todo el desarrollo del algoritmo o durante la ejecución del programa. Es un objeto de valor invariable.  Para expresar una constante se escribe explícitamente su valor.

**Tipos de Constantes:**

Constantes Numéricas (Enteras y Reales)

Constantes Alfanuméricas

Constantes Lógicas (Boolenas)

**Las constantes pueden ser:** Constantes sin nombre: Es una expresión numérica donde se puede utilizar directamente el valor.

Constantes con nombre: Se hace una reserva de memoria en la cual se guarda el valor que será utilizado como constante.

**Ejemplo:**

a) PI = 3.1416

b) E = 2.718228

c) IVA = 0.13

REALIZAR LA SIGUIENTE CONSULTA LA CUAL DEBE QUEDAR EN LA PÁGINA WEB DENTRO DE LA CARPETA DE INFORMÁTICA Y SUBCARPETA ALGORITMOS

**CONECTIVOS LÓGICOS Y PROPOSICIONES COMPUESTAS**

**EXPRESIONES LÓGICAS Y MATEMÁTICAS**

Al evaluar una expresión lógica siempre obtendremos un valor de tipo lógico, es decir verdadero o falso. En estas expresiones se pueden utilizar dos tipos de operadores:

* **Operadores Relacionales:** se utilizan para comparar valores de dos expresiones que pueden ser del mismo tipo: aritmética, lógicas, de carácter o de cadena.

|  |
| --- |
| **OPERADORES REALCIONALES EN PSEUDOCÓDIGOS** |
| **<** | Menor que |  |  |  |
| **<=** | Menor o igual que |  |  |
| **>** | Mayor que |  |  |  |
| **>=** | Mayor o igual que |  |  |
| **=** | Igual que |  |  |  |
| **<>** | Diferente que |  |  |

**Ejemplo:**

22 > 13 (aritméticas) **V**

22.5 ‹ 3.44 (aritméticas) **F**

C > f (de carácter) **F**

Verdadero = falso (lógica) **F**

**VALORES DE VERDAD:**

**V (Verdadero)** y en las tablas de verdad **(1)**

**F (falso)** y en las tablas de verdad **(0)**

* **Operadores Lógicos**

Un **operador lógico** actúa exclusivamente, sobre valores de expresiones lógicas. Los operadores lógicos son:

|  |
| --- |
| **OPERADORES LÓGICOS EN PSEUDOCÓDIGOS** |
| **Y** | Conjunción |  |  |
| **0** | Disyunción |  |  |
| **NO** | Negación |  |  |
|  |  |  |  |

**CONJUNCIÓN (AND) (y):** se utiliza para conectar dos proposiciones que se deben cumplir para obtener un resultado verdadero.

**Ejemplo**:

El coche enciende cuando tiene gasolina en el tanque (y) tiene corriente la batería

P: el coche enciende

Q: tiene gasolina en el tanque

R: tiene corriente la batería

**DISYUNCIÓN (OR) (o):** se deduce que si al menos una de las dos expresiones es verdadera, el resultado será verdadero

**Ejemplo:**

Una persona puede entrar al cine si compra su boleto **(u)** obtiene un pase

P: entra al cine

Q: compra su boleto

**REALIZAR UN ALGORITMO QUE MUESTRE TODO EL**

**PROCESO QUE DEBO TENER EN CUENTA PARA**

**REALIZAR LA MATRÍCULA DEL SEMESTRE II**

# Algoritmo Cualitativo

## Ejemplo1: Freír Tajadas

1.- Buscar utensilios.

2.- Quitar la cáscara al plátano.

3.- Sacar las tajadas del plátano.

4.- Ingresar a la cocina a prender la estufa.

5.- Colocarle aceite al sartén.

6.- Esperar que el aceite este caliente.

¿Estará caliente el aceite?Pregunta de verificación.

**SI:** ir al paso 7.

**NO:** ir al paso 6.

7.- Colocar las tajadas en el sartén.

8.- Esperar a que doren las tajadas por un lado.

¿Estarán listas las tajadas por este lado?Pregunta de verificación.

**SI:** ir al paso 9.

**NO:** ir al paso 8 (Revisión).

9.- Voltear las tajadas.

10.- Esperar a que se doren las tajadas.

¿Estarán doradas las tajadas?

**SI:** ir al paso 11.

**NO:** ir al paso 10.

11.- Retirar las tajadas.

12.- Colocar las tajadas en una vasija.

13.- Apagar la estufa.

14.- Colocar adicionales a la tajada

 - Queso, guayaba**.**

**SI:** ir al paso 14 y seleccionar

**NO:** ir al paso 16.

15.- Servir.

16.- Salir de la cocina