

# Algoritmos/Pseudocódigos

## Introducción



La computadora no implica sólo una herramienta para solucionar problemas. Podemos ser usuarios finales y/o programadores de sistemas automatizados, usando como herramienta de trabajo la computadora, como materia prima nuestra inteligencia, capacidad de análisis, habilidad para detectar soluciones a problemas y como forma de expresión de esas soluciones, utilizaremos los algoritmos y los lenguajes de programación para crear programas.

**Ing. William Mercado, Msc.**



# FUNDAMENTACIONES ALGORÍTMICAS PARA PROGRAMAS DE COMPUTADORAS



**Lenguaje:** Serie de símbolos para transmitir mensajes (ideas) entre dos entidades diferentes. La transmisión de mensajes se conoce comúnmente como **comunicación**.

**Comunicación.** Proceso complejo que requiere una serie de reglas simples, pero indispensables para poderse llevar a cabo. Las dos reglas principales son las siguientes:

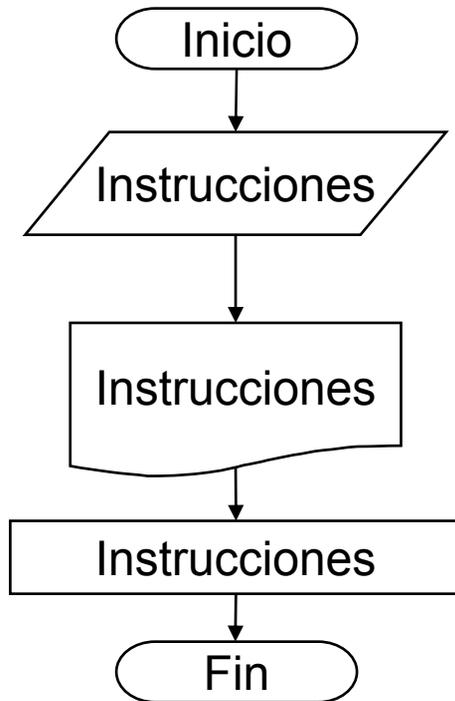
- Los mensajes deben correr en un sentido a la vez.
- Deben forzosamente existir 4 elementos: **Emisor, Receptor, Medio de Comunicación y Mensaje.**

**Algoritmo (Latín árabe alkhwarizmi, nombre de un matemático y astrónomo).**

**Algoritmo.** Serie de pasos organizados que describe el proceso que se debe seguir para dar solución a un problema específico.



## Ejemplo Gráfico



**Lenguajes Algorítmicos:** Conjunto ordenado de símbolos y reglas que se utilizan para describir de manera explícita un proceso.

**Tipos de Lenguajes Algorítmicos:**

**Gráficos:** Representación gráfica de las operaciones que realiza un algoritmo (diagramas de flujo).

**No Gráficos:** Representa en forma descriptiva las operaciones que debe realizar un algoritmo y/o pseudocódigo.

## CARACTERÍSTICAS GENERALES DE UN PROGRAMA O ALGORITMO

### EN GENERAL:

- Debe ser preciso e indicar el orden de realización de cada paso.
- Debe estar definido. Al seguir un algoritmo dos veces, se debe obtener el mismo resultado cada vez.
- Un algoritmo debe ser finito.

**Lenguajes de Programación:** Es un conjunto de símbolos, caracteres y reglas que le permiten a las personas comunicarse con la computadora. Tienen un conjunto de instrucciones que nos permiten realizar operaciones de entrada/salida, cálculo, manipulación de textos, lógica, comparación, almacenamiento, recuperación. Ejemplo: Pascal, Lenguaje C, entre otros...

**OBSERVE COMO ES EL ENTORNO DE TRABAJO DE PASCAL Y TURBO C**



# FUNDAMENTACIONES ALGORÍTMICAS PARA PROGRAMAS DE COMPUTADORAS



```
PROGRAM ALUMNOSPASCAL;  
VAR  
NOMBRE:ARRAY [1..14] OF STRING[30];  
NOTAS:ARRAY[1..6] OF INTEGER;  
PROCEDURE LEERNOMBRESYNOTAS;  
VAR  
I,J:INTEGER;  
BEGIN  
  FOR I:=1 TO 14 DO  
  BEGIN  
    WRITELN ('INGRESE NOMBRE DEL ALUMNO',I);  
    READLN(NOMBRE[I]);  
    FOR J:=1 TO 6 DO  
    BEGIN  
      WRITELN('INGRESE NOTAS DEL ALUMNO',I);  
      READLN(NOTAS[J]);  
    END;  
  END;  
END;  
BEGIN  
END.
```



# FUNDAMENTACIONES ALGORÍTMICAS PARA PROGRAMAS DE COMPUTADORAS



```
for(i=2;i<a1;i++) {
gotoxy(1,i); putchar(' '); gotoxy(an,i); putchar(' '); }
gotoxy(2,1); textattr(AHIIU); cprintf(" vs ",tit); textattr(ANORM);
for(i=0;i<opci[n];i++) dibuja_opcion(n,i,NORMAL); }

int procesar_menu(int n) { int menuy,menux,ancho,alto,tec;
menuy=n+1; menux=2*n+1; ancho=MAXLIT+2; alto=opci[n]+3;
dibuja_menu(n?lite[n-1][sele[n-1]]:"Menu principal",menux,menuy,ancho,alto,n);
for(;;) {
dibuja_opcion(n,sele[n],REVERSE);
tec=getch();
if(tec==ESCAPE) { clrscr(); return(-1); }
if(tec==ENTER) { if(coma[n][sele[n]][0]!='^') {
strcpy(sigmoid,&coma[n][sele[n]][1]); return(1); }
gettext(1,1,80,25,buff);
system("CLS"); system(coma[n][sele[n]]);
getch(); puttext(1,1,80,25,buff); continue; }
if(tec!=ARRIBA&&tec!=ABAJO) continue;
dibuja_opcion(n,sele[n],NORMAL);
if(tec==ARRIBA) { sele[n]--; if(sele[n]<0) sele[n]=opci[n]; continue; }
/* tec==ABAJO */ sele[n]++; if(sele[n]>opci[n]) sele[n]=0; }
return(0); }
```



ACTIVIDAD COMPLEMENTARIA: LEER LA CLASIFICACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN EXPLICADOS EN LA GUÍA DE ESTUDIO. PAG#4

## METODOLOGÍA PARA LA SOLUCIÓN DE PROBLEMAS POR MEDIO DE COMPUTADORAS

**1. Definición del Problema:** Esta fase está dada por el enunciado del problema, el cual requiere una definición clara y precisa. Es importante que se conozca lo que se desea que realice la computadora; mientras esto no se conozca del todo, no tiene mucho caso continuar con la siguiente etapa.



**3. Diseño del Algoritmo:** Debe cumplir con las características de un buen algoritmo, antes citadas.

**4. Codificación**

**6. Documentación**

**7. Mantenimiento**



## ¿Conclusión? Algoritmo

La solución a cualquier problema que involucra la ejecución de una serie de acciones en orden específico. Un procedimiento para resolver un problema en términos de:

- a) Las acciones a ejecutarse y
- b) El orden en el cual estas acciones deben ejecutarse se llama algoritmo.



## ¿CÓMO SE ELABORA UN ALGORITMO SIGUIENDO LOS PASOS ESTUDIADOS?

**A-. DEFINICIÓN DEL PROBLEMA:** ELABORAR UN ALGORITMO QUE SUME DOS NÚMEROS Y LOS MUESTRE POR PANTALLA

**B. ANÁLISIS DEL PROBLEMA:** Entrada: dos números (A,B); Proceso: Operación aritmética (A+B): Suma; Salida: Resultado de la adición

### **C. DISEÑO DEL ALGORITMO:**

1. ALGORITMO SUMA
2. DECLARAR VARIABLES A Y B
3. ESCRIBIR("INTRODUZCA LOS DOS NÚMEROS A SUMAR")
4. LEER(A,B)
5. ESCRIBIR("EL RESULTADO ES:",A+B)
6. FIN ALGORITMO SUMA

### **D. CORRIDA EN FRÍO**

