



PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

ALGORITMO COMPUTACIONAL



ALGORITMO COMPUTACIONAL

Un algoritmo es una secuencia ordenada y finita de instrucciones para realizar un cálculo, desarrollar una tarea o resolver un problema específico. Cada paso en un algoritmo está bien definido y puede ser ejecutado para producir un resultado.

Los algoritmos son la base de la programación y se utilizan para diseñar programas que ejecuten tareas específicas.



Un buen algoritmo debe ser finito, claro y preciso, consistente en sus resultados, eficiente en recursos, general para distintos casos y escalable para manejar mayores volúmenes de datos sin perder rendimiento.





TIPOS DE ALGORITMOS

Búsqueda: localizan un elemento en una colección (como la búsqueda lineal o binaria).

Ordenamiento: organizan datos (como bubble sort o mergesort).

Greedy: eligen la mejor opción en cada paso para lograr una solución óptima (como Huffman o el problema de la mochila).

Programación dinámica: dividen el problema en subproblemas y reutilizan resultados para optimizar cálculos (como Fibonacci optimizado o mochila dinámica).



PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

Ejemplos de algoritmos coloquiales

Se refiere a secuencias de pasos que usamos en la vida cotidiana para resolver problemas o realizar tareas, expresadas en lenguaje común (no en código).



Algoritmo para planchar una camisa

1. Abrir la tabla de planchar
2. Rellenar la plancha de agua y encenderla
3. Colocar la camisa desabotonada sobre la tabla de planchar
4. Planchar la parte de los hombros y la espalda superior
5. Planchar de derecha a izquierda sin tocar las mangas
6. Planchar los brazos de la camisa
7. Sacar una percha
8. Colgar la camisa en la percha
9. Abrir el armario
10. Guardar la camisa en el armario
11. Cerrar el armario



Algoritmo para ver una película

1. Encender el televisor, el ordenador o el dispositivo desde el cual queremos ver la película
2. Entrar en internet
3. Abrir nuestra plataforma de contenido de entretenimiento de referencia
4. Hacer clic en el símbolo de la lupa para hacer una búsqueda exacta
5. Teclear el nombre de la película que queremos ver
6. Seleccionar la película
7. Hacer clic en el botón de inicio para comenzar el visionado



Ejemplos de algoritmos en PSeInt

PSeInt es una herramienta educativa diseñada para ayudar a aprender lógica de programación y estructuras algorítmicas de manera sencilla.



PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

Su nombre significa “Pseudo Intérprete”, porque permite escribir algoritmos en un pseudocódigo, es decir, un lenguaje parecido al español que simula la forma en que se programa, pero sin necesidad de conocer aún un lenguaje de programación real



Tipos de datos básicos

Debes saber que cada variable puede guardar un tipo de información:

- **Entero:** números sin decimales (ej. 10, -3).
- **Real:** números con decimales (ej. 4.5, -2.8).
- **Caracter:** letra, texto (ej. "A", "Hola").
- **Lógico:** valores Verdadero o Falso.



Operaciones básicas

- **Aritméticas:** +, −, *, /, ^ (potencia).
- **Relacionales:** =, <>, <, >, <=, >=.
- **Lógicas:** Y, O, NO (para condiciones).



**UNIVERSIDAD
DEL PACIFICO**

PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

Estructuras fundamentales

Estas son la base de cualquier algoritmo:

Entrada y salida de datos:

- Leer nombre
- Escribir "Hola ", nombre



Cómo declarar variables

Antes de usarlas, siempre define qué tipo de dato guardarán:

Definir edad Como Entero

Definir nombre Como Cadena



Condicionales (si...entonces):

```
Si edad >= 18 Entonces  
    Escribir "Eres mayor de edad"  
SiNo  
    Escribir "Eres menor de edad"  
FinSi
```



Bucle “Para” (contador definido)

Se usa cuando sabes cuántas veces debe repetirse la acción.

```
Para i <- 1 Hasta 5 Con Paso 1 Hacer  
    Escribir "Número ", i  
FinPara
```



**UNIVERSIDAD
DEL PACIFICO**

PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

Bucle “Mientras” (condición al inicio)

Se ejecuta mientras la condición sea verdadera.
Se usa cuando no sabes cuántas veces se repetirá.

```
contador <- 1
```

```
Mientras contador <= 5 Hacer  
  Escribir "Contando: ", contador  
  contador <- contador + 1  
FinMientras
```



Bucle “Repetir” (condición al final)

Se ejecuta al menos una vez, y se repite hasta que la condición sea verdadera.

numero <- 1

Repetir

 Escribir "Número actual: ", numero

 numero <- numero + 1

Hasta Que numero > 5



**UNIVERSIDAD
DEL PACÍFICO**

PROGRAMA: INGENIERIA DE SISTEMAS
SEMESTRE: PRIMERO
DOCENTE: JAMINTON ASPRILLA A.

PSeInt

Archivo Editar Configurar Ejecutar Ayuda

Operadores y Funciones

[-] Op. Algebraicos

- + (suma)
- (resta)
- * (multiplicación)
- / (división)
- ↑ (potencia)
- MOD (resto/modulo)

[-] Op. Lógicos

- Y (conjunción)
- O (disyunción)
- NO (negación)

[-] Op. Relacionales

- = (igual)
- ≠ (distinto)
- < (menor)
- ≤ (menor o igual)
- > (mayor)
- ≥ (mayor o igual)

[-] Func. Matemáticas

- abs (valor absoluto)
- trunc (valor truncado)
- redon (valor redondeado)
- raiz (raíz cuadrada)
- sen (seno)
- cos (coseno)
- tan (tangente)
- asen (arcoseno)
- acos (arcocoseno)
- atan (arcotangente)
- ln (logaritmo natural)
- exp (func. exponencial)

[-] Func. p/Cadenas

- Longitud
- SubCadena
- Concatenar

<sin_titulo>

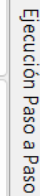
```
1 Algoritmo sin_titulo
2
3 FinAlgoritmo
4
```

Comandos

- Escribir
- Leer
- Asignar
- Si-Entonces
- Según
- Mientras
- Repetir
- Para
- Función

El pseudocódigo es correcto. Presione F9 para ejecutarlo.

Comandos



Consejo para estudiantes de primer semestre que inician con algoritmos en PSeInt

Estimados estudiantes, comenzar con algoritmos es dar el primer paso hacia pensar como programadores.

Antes de preocuparse por el código o los comandos, aprendan a razonar paso a paso:

>> todo algoritmo es simplemente una forma ordenada de resolver un problema.



Algunos consejos prácticos:

- 1. Piensen antes de escribir:** dibujen el proceso, hagan un diagrama o escriban los pasos en palabras simples antes de usar PSeInt.
- 2. Empiecen con problemas pequeños:** sumas, promedios o comparaciones. La práctica constante los llevará a dominar estructuras más complejas.
- 3. Comprendan las estructuras básicas:** secuencias, decisiones (Si...Entonces) y repeticiones (Mientras, Para, Repetir). Son la base de toda programación.
- 4. Lean los mensajes de error:** PSeInt es una herramienta para **aprender**, no solo para ejecutar. Cada error es una oportunidad de entender mejor la lógica.
- 5. Trabajen en equipo:** explicar un algoritmo a otro compañero les ayudará a reforzar su comprensión.
- 6. Sean pacientes y curiosos:** programar no es memorizar comandos, es **aprender a pensar de manera lógica y creativa**.

