

# “UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE”



## COMPLEJIDAD ALGORÍTMICA

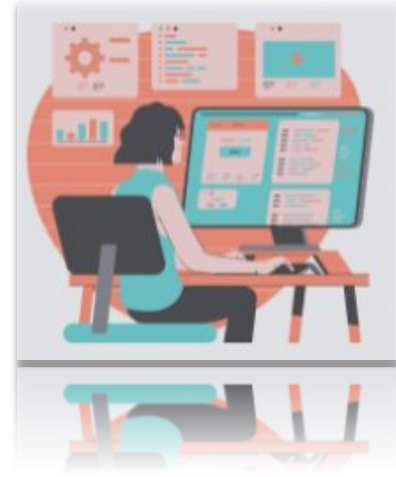
### INTEGRANTES:

- CARLOS IPIALES
- RICARDO GRIJALVA
- FERNANDO TIPAN
- KARLA CAJAS



# COMPLEJIDAD ALGORÍTMICA

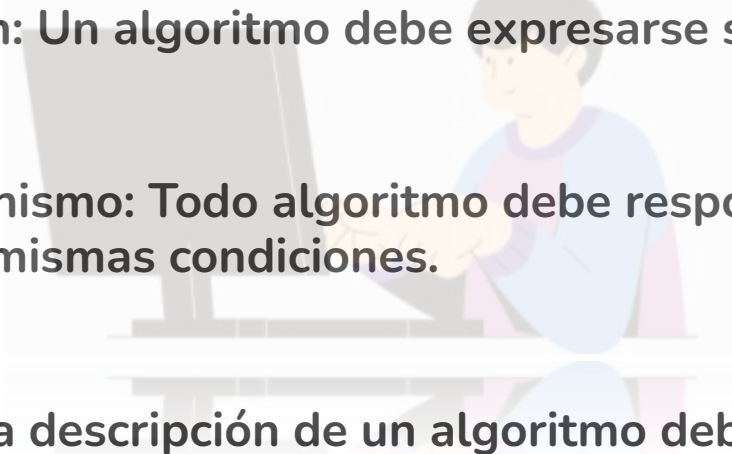
La complejidad algorítmica representa la cantidad de recursos (temporales) que necesita un algoritmo para resolver un problema y por tanto permite determinar la eficiencia de dicho algoritmo.





# CARACTERÍSTICAS DE UN ALGORITMO

Un algoritmo debe poseer las siguientes características:

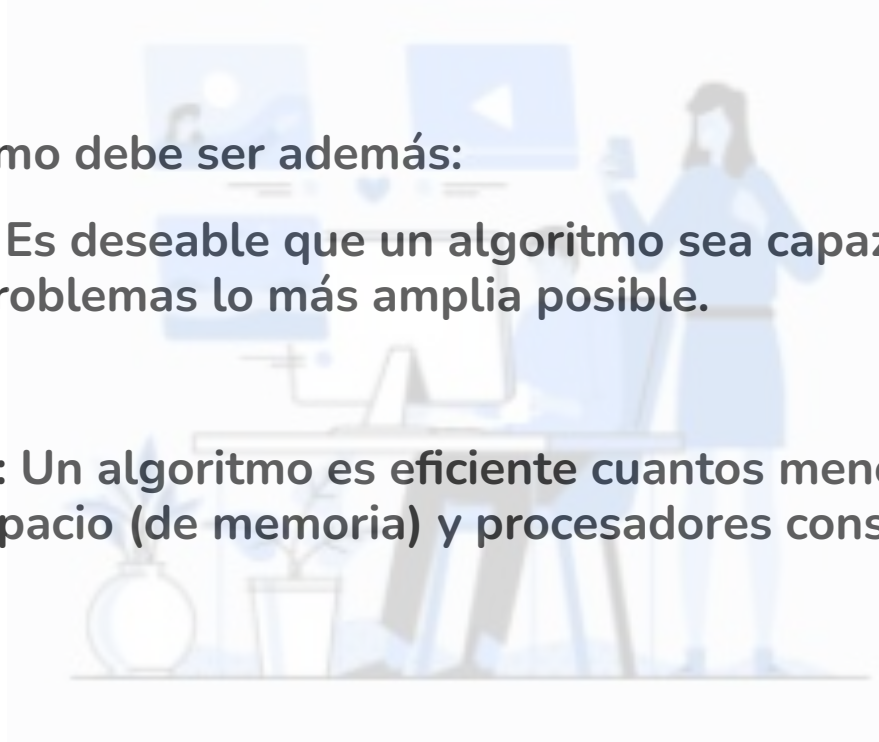
- Precisión: Un algoritmo debe expresarse sin ambigüedad.
  - Determinismo: Todo algoritmo debe responder del mismo modo antes las mismas condiciones.
  - Finito: La descripción de un algoritmo debe ser finita.
- 



# CARACTERÍSTICAS DE UN ALGORITMO

Un algoritmo debe ser además:

- General: Es deseable que un algoritmo sea capaz de resolver una clase de problemas lo más amplia posible.
- Eficiente: Un algoritmo es eficiente cuantos menos recursos en tiempo, espacio (de memoria) y procesadores consume





# EVALUACIÓN DE COSTE PARA COMPLEJIDAD ALGORÍTMICA

El tiempo empleado por el algoritmo se mide en pasos.

- El coste depende del tamaño de los datos.
- A la hora de evaluar el coste se debe de tener en consideración tres posibles casos:
  - El coste esperado o promedio
  - El coste mejor
  - El coste peor



# **EVALUACIÓN DE COSTE PARA COMPLEJIDAD ALGORÍTMICA**

El tiempo empleado por el algoritmo se mide en pasos.

- El coste depende del tamaño de los datos.
- A la hora de evaluar el coste se debe de tener en consideración tres posibles casos:
  - El coste esperado o promedio
  - El coste mejor
  - El coste peor



# EVALUACIÓN DE COSTE PARA COMPLEJIDAD ALGORÍTMICA

EL COSTE EN TIEMPO DEPENDE DEL TAMAÑO DE LOS DATOS

El tiempo requerido por una algoritmo es función del tamaño de los datos.

- Por esta razón la complejidad temporal se expresa de la siguiente forma:

$$T(n)$$

EL COSTE ESPERADO, EL MEJOR Y EL PEOR. NOTACIÓN

- $T_{\max}(n)$

- $T_{\min}(n)$

- $T_{\text{med}}(n)$

# TIPOS

## COMPLEJIDAD TEMPORAL

- Es la complejidad computacional que describe la cantidad de tiempo que lleva ejecutar un algoritmo.
- La complejidad temporal se estima comúnmente contando el número de operaciones elementales realizadas por el algoritmo, suponiendo que cada operación elemental requiere una cantidad fija de tiempo. Por lo tanto, la cantidad de tiempo necesario y el número de operaciones elementales realizadas por el algoritmo difieren en un factor constante como máximo.







# COMPLEJIDAD ESPACIAL

Memoria que utiliza programa para su ejecución . la eficiencia de memoria un algoritmo la cantidad de espacio requerido para ejecutar el algoritmo ; es decir , el espacio de memoria que ocupan todas las variables propias al algoritmo

- La cantidad de memoria requerida por el código del algoritmo.
- La cantidad de memoria requerida para almacenar los datos de entrada.
- La cantidad de memoria requerida para los datos de salida (algoritmos como los de ordenación suelen organizar los datos de entrada y por ello no necesitan memoria extra para la salida).

# COMPLEJIDAD SINTÁCTICA

Consiste el cálculo de complejidad temporal a prioridad de un algoritmo de función del tamaño del problema  $n$ , presidiendo de factores constantes multiplicativos y suponiendo valores de  $n$  muy grandes



# TIEMPO DE EJECUCIÓN

El tiempo de ejecución de un programa se da en función  $N$  lo que designaremos  $N(T)$

Esta función se puede calcular físicamente ejecutando el programa acompañado de un reloj o calcular directamente sobre el código, contando las instrucciones a ser ejecutadas y multiplicar por el tiempo requerido por cada instrucción.

