



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

Tema:

# Tiempo de ejecución de un algoritmo

---

GRUPO 2 - GRUPO 12

**Autores:**

Richard Albán

Juan Gallardo

Eduardo Mortensen

**Materia:**

Estructura de datos

**NRC:**

7166





# ÍNDICE

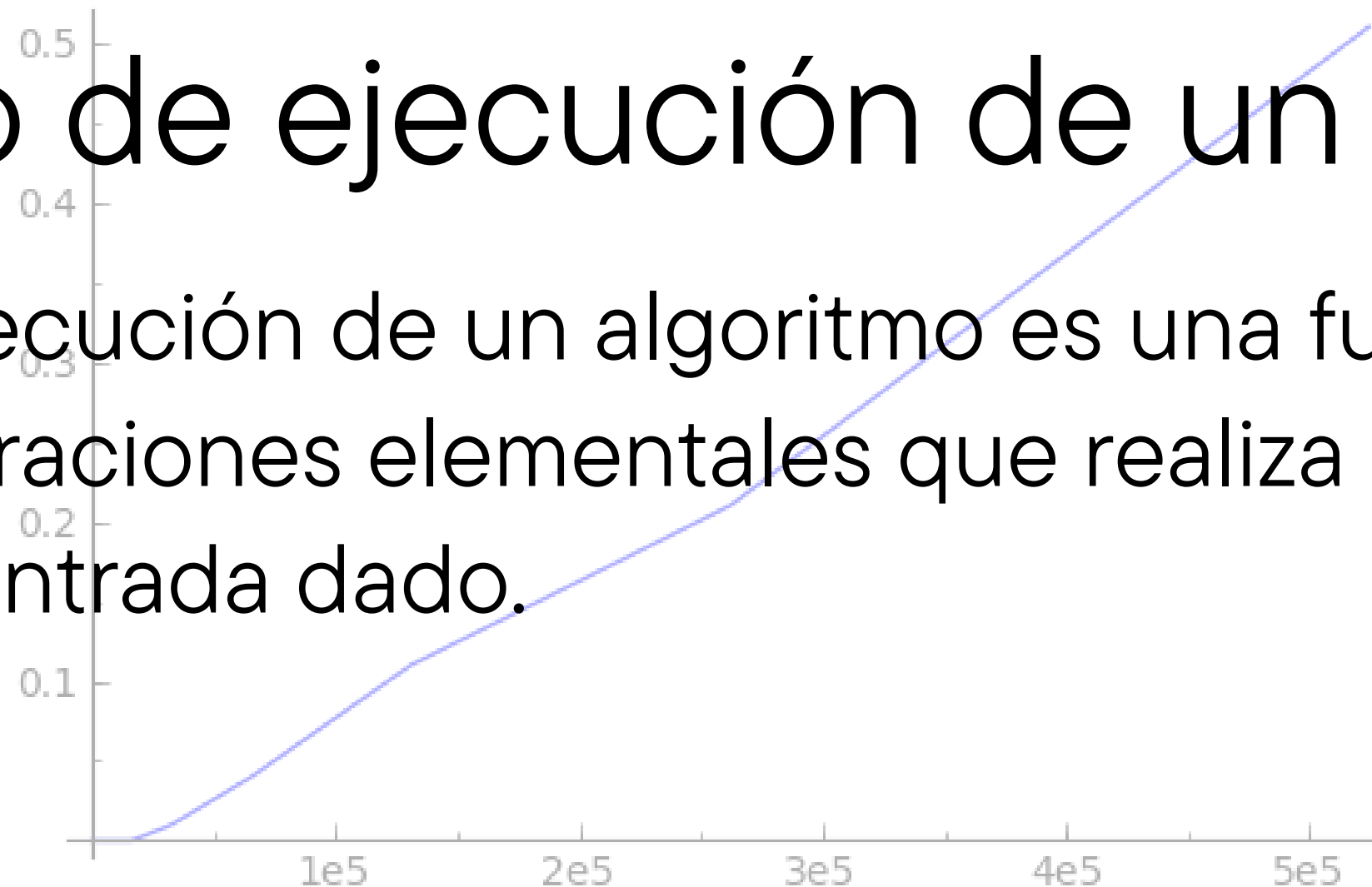
## TIEMPO DE EJECUCIÓN DE UN ALGORITMO

- Concepto
- Características
- Eficiencia
- Cómo medir
- Importancia
- Ejemplos
- Conclusiones

# ANÁLISIS DE ALGORITMOS

## Tiempo de ejecución de un algoritmo

El tiempo de ejecución de un algoritmo es una función que mide el número de operaciones elementales que realiza el algoritmo para un tamaño de entrada dado.



# CARACTERISTICAS DEL TIEMPO DE EJECUCIÓN DE UN ALGORITMO

**Estos criterios se centran principalmente en su simplicidad y en el uso eficiente de los recursos:**

- La sencillez es una característica muy interesante a la hora de diseñar un algoritmo, pues facilita su verificación, el estudio de su eficiencia y su mantenimiento.
- De ahí que muchas veces prime la simplicidad y legibilidad del código frente a alternativas más crípticas y eficientes del algoritmo.

EFICIENCIA DEL TIEMPO DE EJECUCIÓN DE UN ALGORITMO

RESPECTO AL USO EFICIENTE DE LOS RECURSOS

**Al evaluar un algoritmo se debe tener en cuenta su eficiencia en cuanto a:**

- Consumo de memoria
- Tiempo de ejecución

Teniendo en cuenta el tamaño de la entrada de datos

# ¿CÓMO MEDIR EL TIEMPO DE EJECUCIÓN DE UN ALGORITMO?

Existen dos formas: a posteriori y a priori

## **Posteriori**

Solución del problema (Identificación de requisitos, abstracción de clases, escritura de pseudocódigo).

## **Priori**

La evaluación del algoritmo se realiza con independencia del lenguaje de programación en el cual se codifique y en donde se ejecute

## Midiéndolo en función de N:

El tiempo de ejecución de un programa se mide en función de N, lo que designaremos como  $T(N)$ . Habitualmente todos los algoritmos contienen alguna sentencia condicional, haciendo que las sentencias ejecutadas dependan de la condición lógica.

```
/*Esta función se puede calcular el tiempo de ejecución  
del programa acompañado de un reloj*/  
/* (int x = 0; x < N; x++ */  
for (int i = 0; i < 1000; i++)
```

Y va a tardar:

```
/*Donde "fin" es el tiempo que lleva ejecutar la serie 1 de sentencias  
y el "inicio" la serie 2*/  
fin = clock();  
t = fin - inicio;
```



## EXPLICADO DETALLADAMENTE:

```
void tiempoProc1(void)
{
    int inicio, fin, t;
    float tiempoSeg;
    inicio = clock(); //guarda el tiempo inicial
    for (int i = 0; i<1000; i++)
        cout <<i<<"n";
    fin = clock(); //guarda tiempo final
    t = fin - inicio; //calcula tiempo ptranscurrido
    tiempoSeg=((float)t); //el número de tics del reloj por segundo
    cout << "El tiempo transcurrido fue: " << tiempoSeg << "n";
}
```





En C++ hay varias formas de hacerlo, pero dependiendo de cómo lo implementemos y qué cabeceras usemos

```
#include <iostream>
#include <ctime>
#include <time.h>
#include <math.h>
```

# IMPORTANCIA

REL TIEMPO DE EJECUCIÓN DE UN ALGORITMO VA A SER UNA FUNCIÓN QUE MIDE EL NÚMERO DE OPERACIONES ELEMENTALES QUE REALIZA EL ALGORITMO PARA UN TAMAÑO DE ENTRADA DADO.



# EJEMPLOS