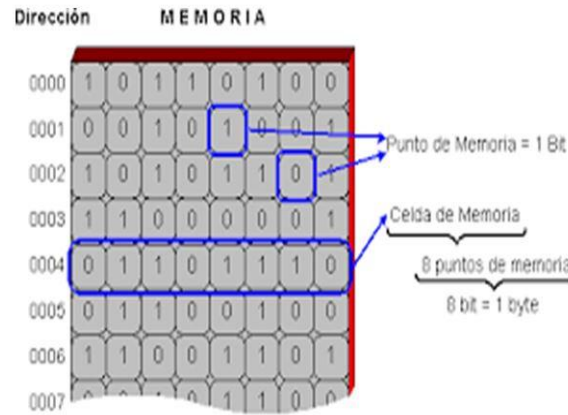
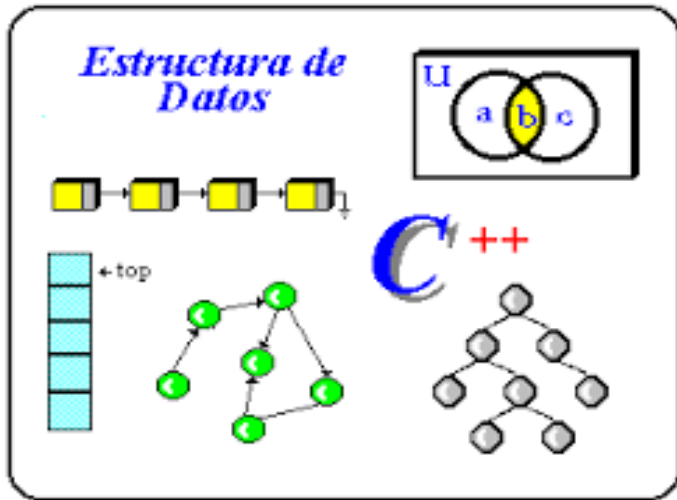


# Estructuras de



## Datos en Memoria RAM



## Algoritmos y Programación

Cálculo de Complejidad Espacial

# Complejidad en Algoritmos

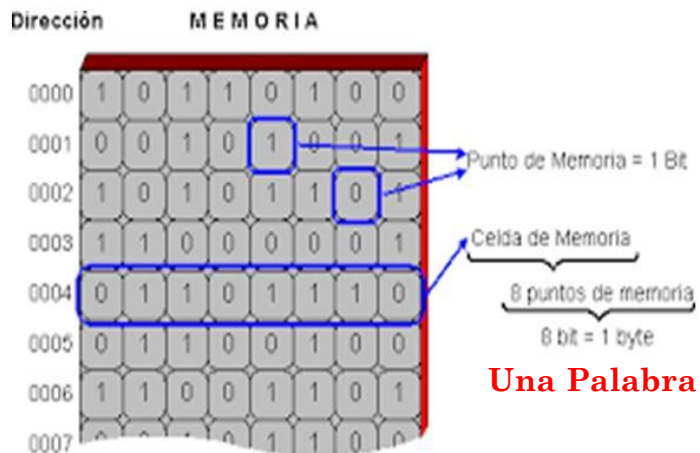
## Complejidad Espacial

La complejidad espacial es la cantidad de espacio en memoria que un algoritmo emplea al ejecutarse. En otras palabras, cómo el algoritmo ocupa espacio en memoria con la cantidad de elementos de entrada que debe procesar.

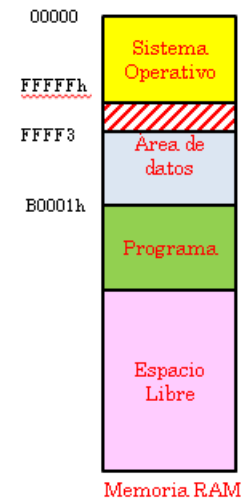
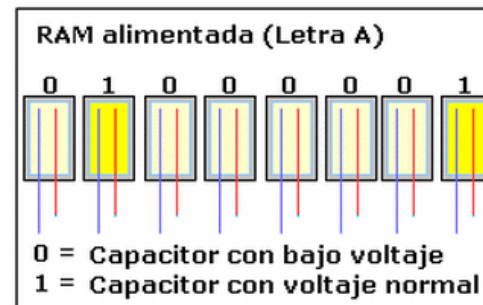
## Elementos relacionados con la Complejidad Espacial

La Complejidad Espacial esta relacionada con TODAS las Estructuras de Datos que utiliza el Algoritmo y el tipo primitivo almacenado las cuales tienen asignado un espacio en Memoria RAM el cual puede variar entre un Lenguaje de Programación y otro.

## Forma de Almacenamiento y Medidas de las Estructuras en Memoria RAM



**Una Palabra**



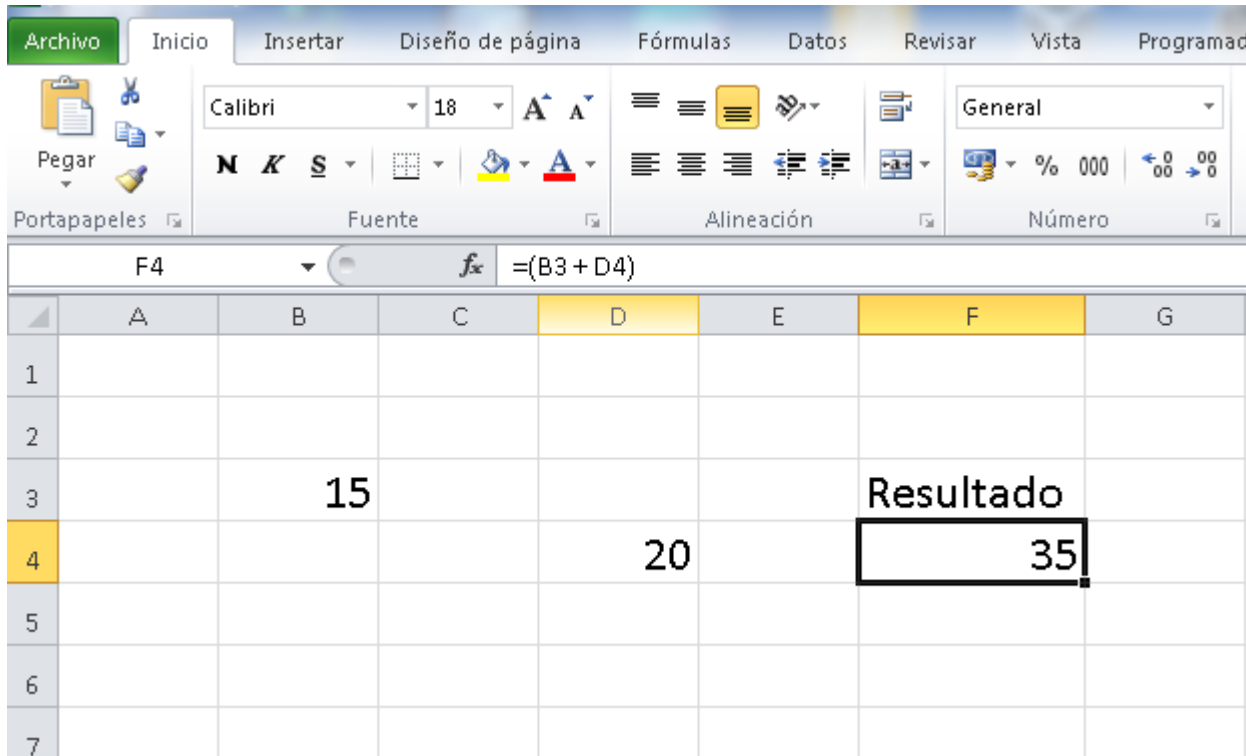
# Complejidad en Algoritmos

## Complejidad Espacial

### Organización de la Memoria RAM

Entendiendo la forma de su Organización

**Ejemplo** ( Analogía con los programas de Hojas de Cálculo (Excel) )

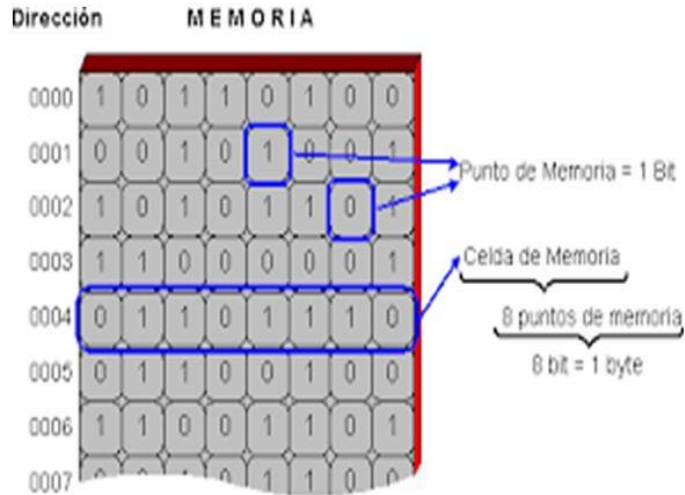


	A	B	C	D	E	F	G
1							
2							
3		15				Resultado	
4				20		35	
5							
6							
7							

# Complejidad en Algoritmos

## Complejidad Espacial

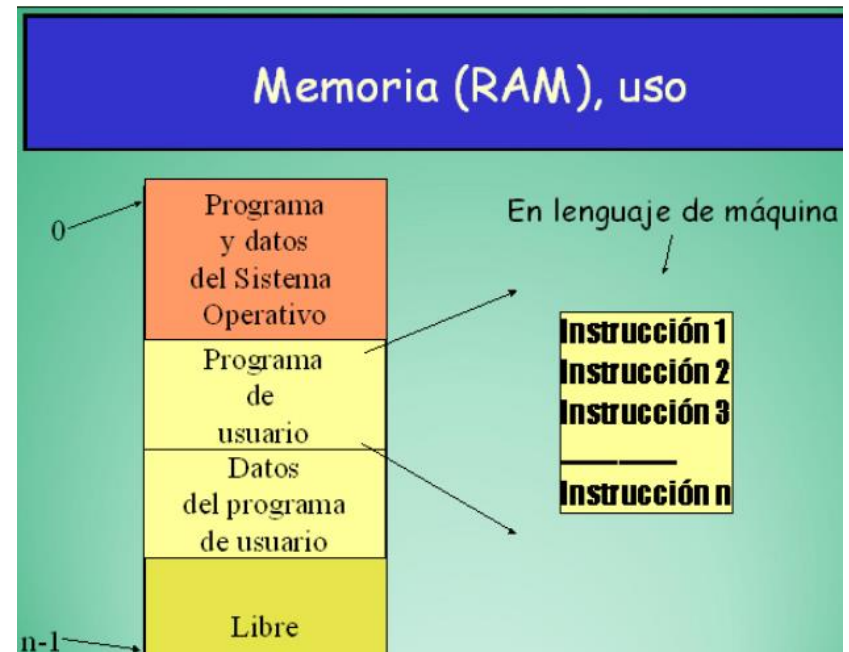
### Organización de la Memoria RAM



### Principales características de la Memoria RAM

- La Memoria esta dividida en celdas de igual tamaño
- Cada celda de memoria tiene asociado un número o dirección(hexadecimal) que la identifica
- La Celdas de memoria RAM se pueden leer y escribir.
- Las Celdas siempre tiene valores

- Las combinaciones de 0 y 1 pueden representar todo tipo de datos
- Si se apaga el computador los datos almacenados en la memoria (RAM) se pierden
- Solo se almacena un dato a la vez



## Complejidad Espacial

### Que son las Estructuras de Datos

Son las diferentes alternativas para almacenar y manipular los datos en la Memoria RAM

### Tipos de Estructuras de Datos

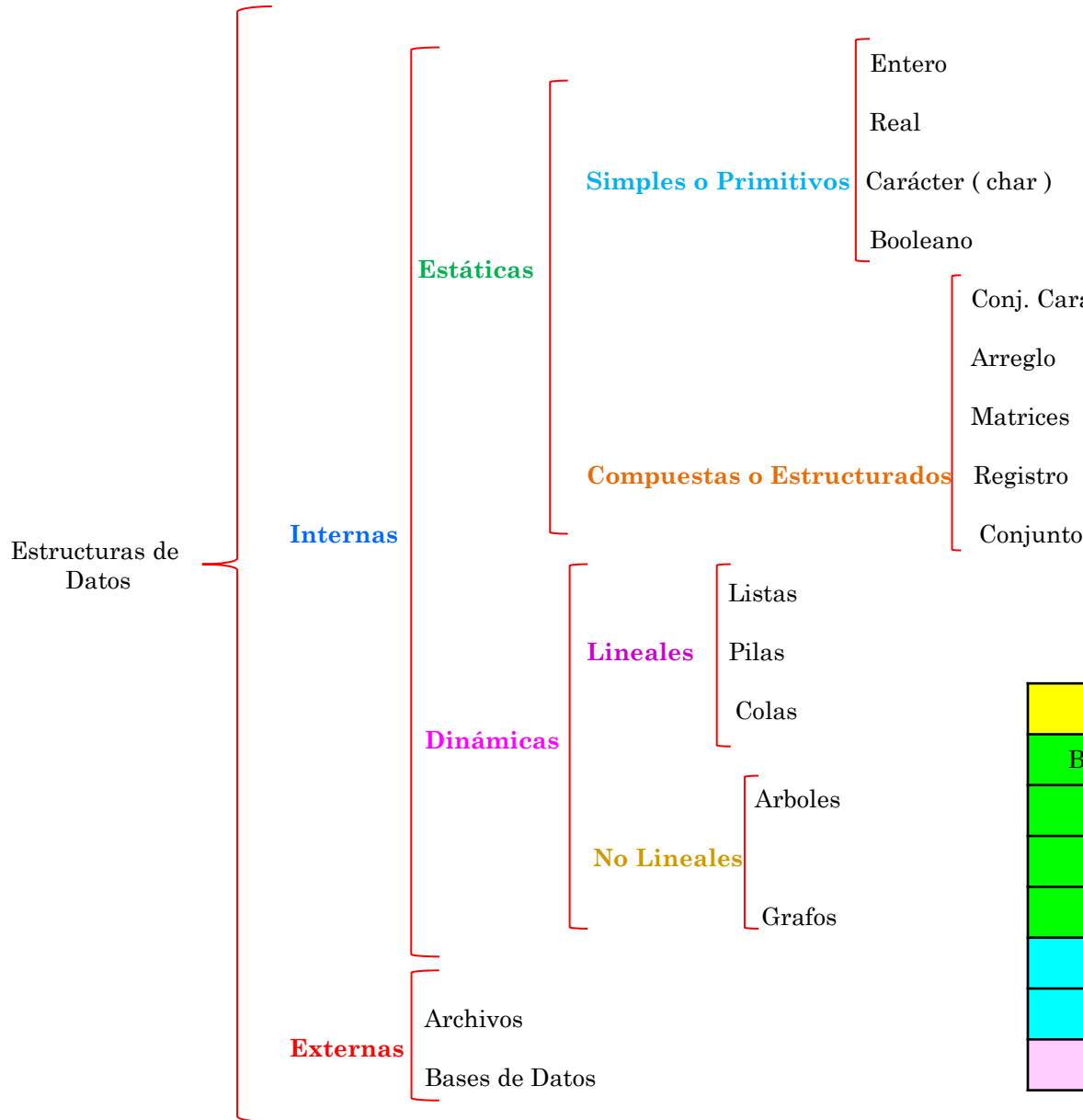
Las Estructuras de Datos pueden ser de diferentes tipos, dependiendo de la técnica que se utilice, para su almacenamiento y recuperación, en donde cada una proporciona características idóneas para el correcto y eficiente tratamiento de la información que en ellas se almacena.

Las diferencias principales entre cada una de ellas radican en:

- .- La Procedencia de los datos que almacena.
- .- Su Capacidad de Almacenamiento y tipos de datos que permite.
- .- Como están Conformadas
- .- Las formas de hacer referencia a los valores que almacenan y formas de recorrerlas

# Complejidad en Algoritmos

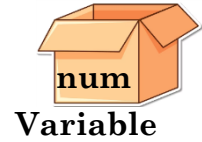
## Complejidad Espacial



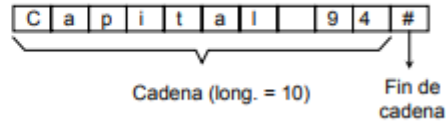
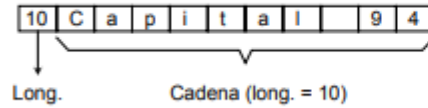
**Variables**  
**Tipos de Datos Primitivos**  
**Espacio reservado**

Tipo	Tamaño en Bytes
Byte, Char, Boolean	1 Byte
Short	2 Byte
Int	4 Bytes
Long	8 Bytes
Float	4 Bytes
Double	8 Bytes
String	4 Bytes

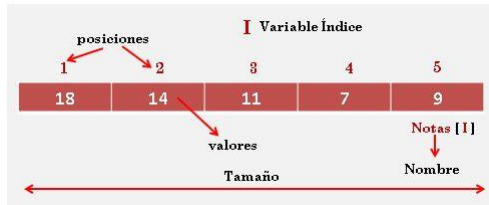
## Variable



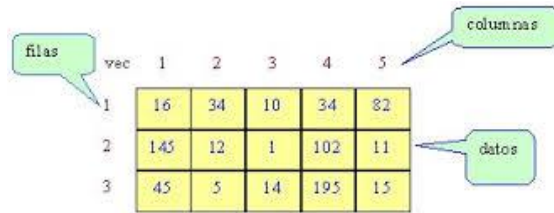
## Cadenas o String



## Arreglos



## Matrices



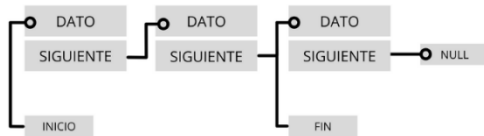
## Registros



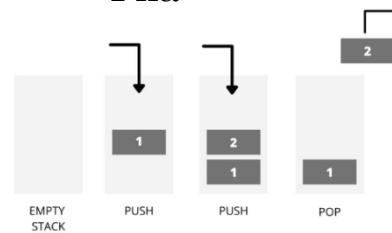
## Conjuntos



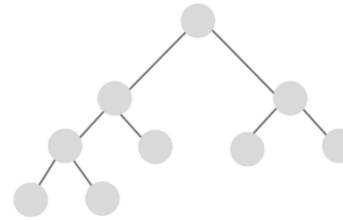
## Lista



## Pila



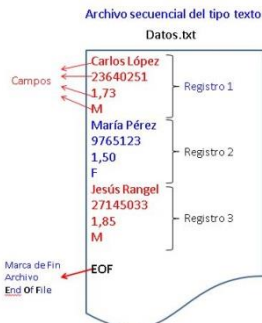
## Arboles



## Grafos



## Archivos



## BASES DE DATOS



# Complejidad en Algoritmos

## Complejidad Espacial

### Ejemplo

Dado un arreglo de 10 posiciones llamado N, el cual contiene números enteros positivos, se desea elaborar un programa en Python que:

- Almacene en el arreglo números comprendidos entre 10 y 50 de forma aleatoria y muéstrelo
- Determine cual es el Mayor número indicando su posición
- Ordene el Arreglo en orden ascendente
- Determine cual es el Mayor número indicando su posición
- Busque un número dado por el usuario en el arreglo indicando con un mensaje si este se encuentra y mostrando además su posición
- Calcule la Complejidad Temporal aplicando la Notación Big O del Algoritmo indicando finalmente el Orden de esta.
- Calcule la Complejidad Espacial del Algoritmo.