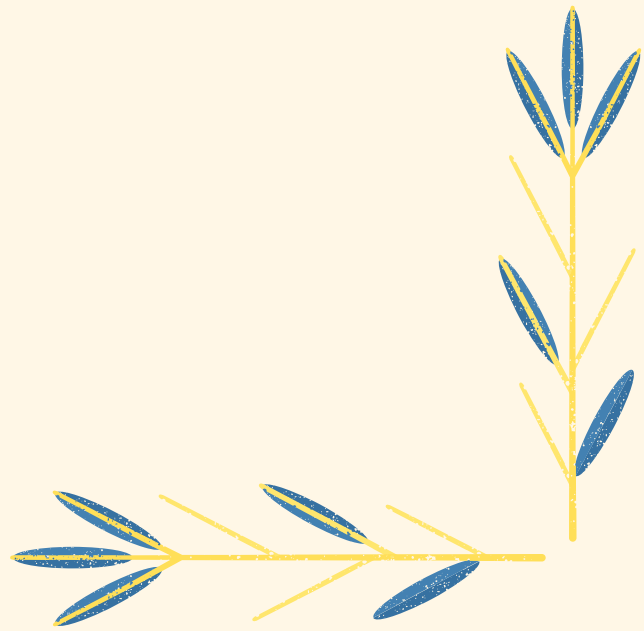


Estructuras De Datos

Gustavo
Vitola



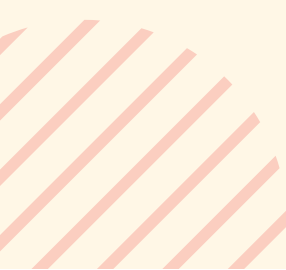
¿Qué son?



Forma particular de organizar información en un computador para que pueda ser utilizada de manera eficiente

Diferentes tipos de estructuras de datos son adecuados para diferentes tipos de aplicaciones, y algunos son altamente especializados para tareas específicas.

Son medios para manejar grandes cantidades de información de manera eficiente para usos tales como grandes bases de datos y servicios de indización de Internet



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, estos tipos son los siguientes:

- Estructura de datos estática.
- Estructura de datos dinámica.

Según la secuencia que se presenta entre cada elemento al momento de realizar el recorrido entre los elementos de la estructura de datos, esta se puede clasificar en los siguientes tipos:

- Estructura de datos lineal.
- Estructura de datos no lineal.

Estructura de datos lineales

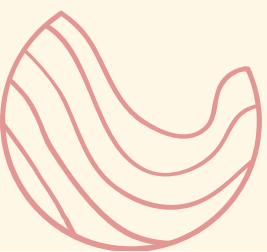
Las estructuras de datos lineales son aquellas en las que los elementos ocupan lugares sucesivos en la estructura y cada uno de ellos tiene un único sucesor y un único predecesor, es decir, sus elementos están ubicados uno al lado del otro relacionados en forma lineal.

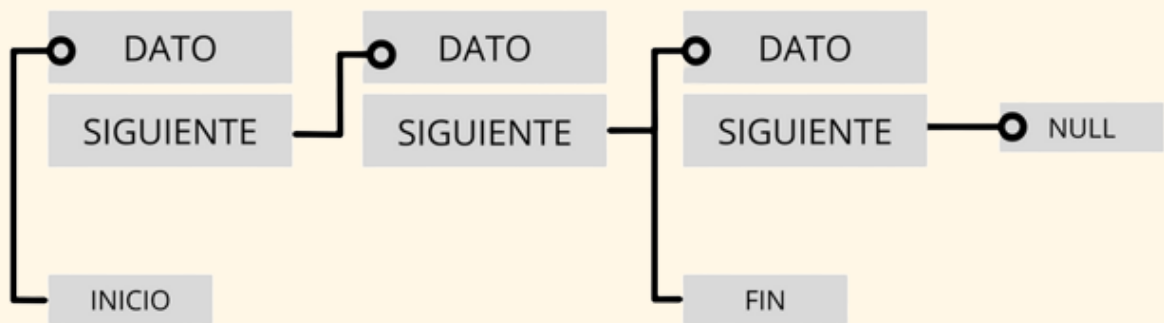
Hay tres tipos de estructuras de datos lineales:

- Listas enlazadas
- Pilas
- Colas

Listas enlazadas

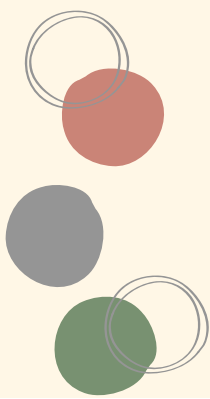
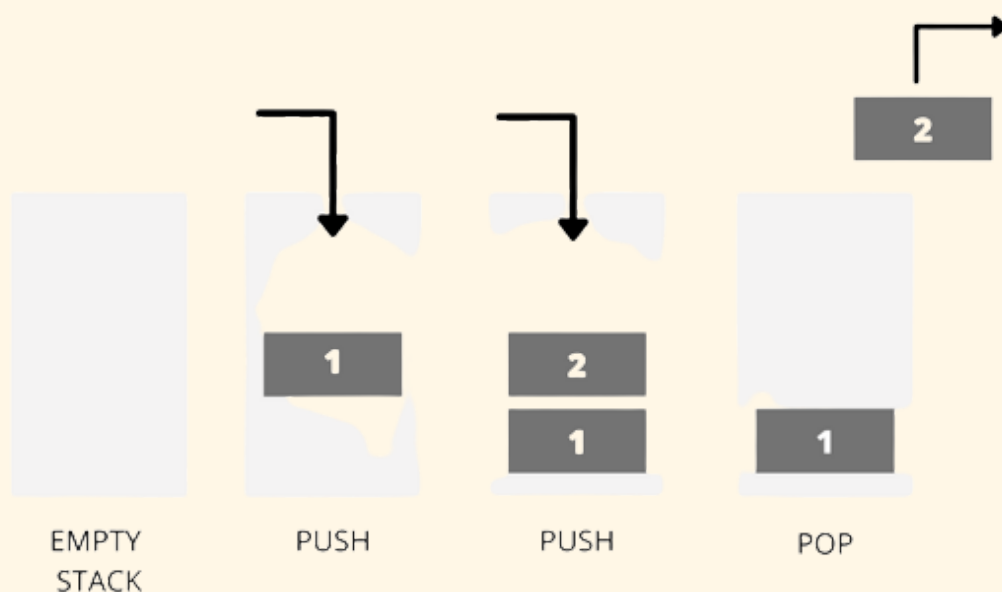
En las estructuras de datos, las listas enlazadas se construyen con elementos que están ubicados en una secuencia. Aquí, cada elemento se conecta con el siguiente a través de un enlace que contiene la posición del siguiente elemento. De este modo, teniendo la referencia del principio de la lista podemos acceder a todos los elementos de la misma.





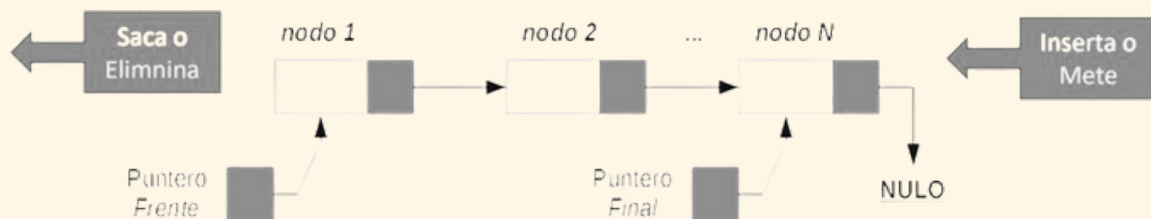
Pila

La pila es un tipo especial de lista lineal dentro de las estructuras de datos dinámicas que permite almacenar y recuperar datos, siendo el modo de acceso a sus elementos de tipo LIFO (del inglés Last In, First Out, es decir, último en entrar, primero en salir)



Cola

Es una estructura de datos que almacena elementos en una lista y permite acceder a los datos por uno de los dos extremos de la lista. Un elemento se inserta en la cola (parte final) de la lista y se suprime o elimina por la frente (parte inicial, cabeza) de la lista.



Estructura de datos no lineales

Las estructuras de datos no lineales, también llamadas multienlazadas, son aquellas en las que cada elemento puede estar enlazado a cualquier otro componente. Es decir, cada elemento puede tener varios sucesores o varios predecesores.

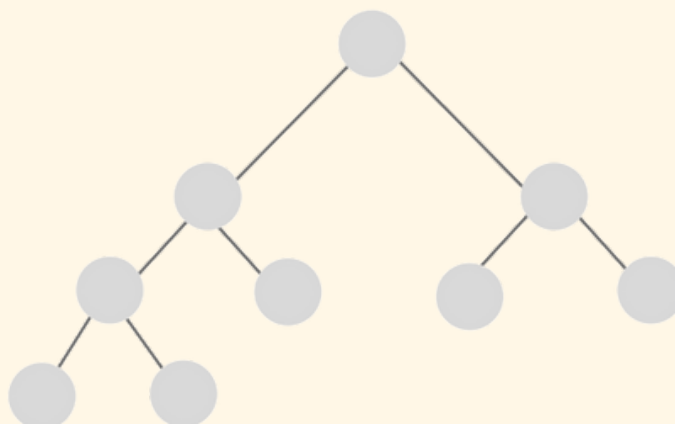
Existen dos tipos:

Árboles

Grafos

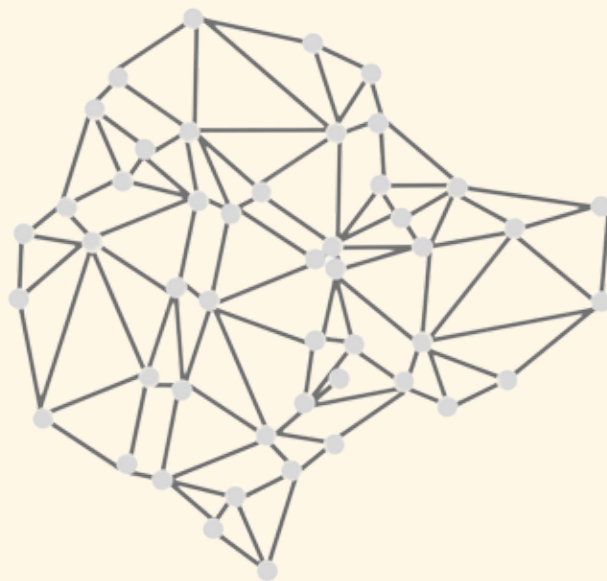
Árboles

En estructura de datos, los árboles consisten en una estructura no lineal que se utiliza para representar datos con una relación jerárquica en la que cada elemento tiene un único antecesor y puede tener varios sucesores.



Grafos

Otro tipo de no lineal de estructura de datos en programación, son los grafos. Se trata de una estructura matemática formada por un conjunto de puntos —una estructura de datos— y un conjunto de líneas, cada una de las cuales une un punto a otro. Los puntos se llaman nodos o vértices del grafo y las líneas se llaman aristas o arcos.



Estructuras de datos en Python

Las estructuras de datos que nos ofrece este maravilloso lenguaje de programación de forma nativa:

Listas

Tuplas

Sets

Strings

Diccionarios

Listas

Las listas o arrays en python son estructuras de datos muy flexibles en las que podemos mezclar valores de varios tipos, o bien que sean de un solo tipo. Su declaración es sencilla y obedece un formato JSON estándar.

```
>>> [1, 2, 3, 4]
```

```
[1, 2, 3, 4]
```

```
>>> ["hola", "mundo"]
```

```
["hola", "mundo"]
```

```
>>> [0, 1.5, "hola"]
```

```
[0, 1.5, "hola"]
```

```
>>> [0, 1.5, "hola"]
```

```
[0, 1.5, "hola"]
```



Tuplas

Las tuplas son como las listas, excepto que en este caso son inmutables. Una tupla consiste de varios valores separados por comas.

```
>>> a = (1, 2, 3)
>>> a[0]
1
```

Sets

Los sets en python son listas sin un orden específico pero cuyos elementos son únicos, es decir, no existe la repetición. Y para definir uno hay que explícitamente indicar que queremos que la lista sea un set con la función con el mismo nombre,

```
>>> x = set([3, 1, 2, 1])
set([1, 2, 3])
```



Strings

Una de las cosas curiosas sobre python, es que en muchas maneras podemos manipular strings de la misma manera como se manipulan las listas. Esto convierte a los strings en una especie de estructura de datos conveniente de usar.

Podemos usar len en un string

```
>>> len("abrakadabra")
```

11



Diccionarios

Un diccionario en python actua de manera similar a una lista, excepto que el indice de este no necesariamente tiene que ser un numero entero, y se asemejan bastante a un objeto JSON

```
>>> a = {'x': 1, 'y': 2, 'z': 3}
```

```
>>> a['x']
```

```
1
```

```
>>> a['z']
```

```
3
```

```
>>> b = {}
```

```
>>> b['x'] = 2
```

```
>>> b[2] = 'foo'
```

```
>>> b[(1, 2)] = 3
```

```
>>> b
```

```
{(1, 2): 3, 'x': 2, 2: 'foo'}
```

