

Estructuras de Datos

Definición

- Representación, organización y almacenamiento de información en la computadora.
- Diferentes tipos según diferentes aplicaciones/necesidades.
- Sirven para manejar grupos de datos.
- Cada lenguaje las implementa de forma diferente.

Estructuras de datos básicas

Listas

- Se les conoce con diferentes nombres según el lenguaje de programación: arreglos, vectores.
- Por lo general, las listas se componen de elementos del mismo tipo.
- Python no tiene esa restricción :-)

```
In [1]: lista = [10, "hola", True]
```

```
In [2]: lista
```

```
Out[2]: [10, 'hola', True]
```

```
In [3]: type(lista)
```

```
Out[3]: list
```

```
In [4]: for x in lista:  
...:     print type(x)  
...:
```

```
<type 'int'>
```

```
<type 'str'>
```

```
<type 'bool'>
```

Listas

- Los arreglos o listas se accesan por posición de los elementos.
- En la mayoría de lenguajes (incluido Python) la primera posición es la 0, y la última es la ***n - 1***

```
In [52]: len(lista)
Out[52]: 3

In [53]: print lista[0], lista[1], lista[2]
10 hola True

In [54]: print lista[3]
-----
IndexError                                Traceback (most recent call last)
/home/jeudy/Proyectos/UCR/<ipython-input-54-d61322994e32> in <module>()
----> 1 print lista[3]

IndexError: list index out of range
```

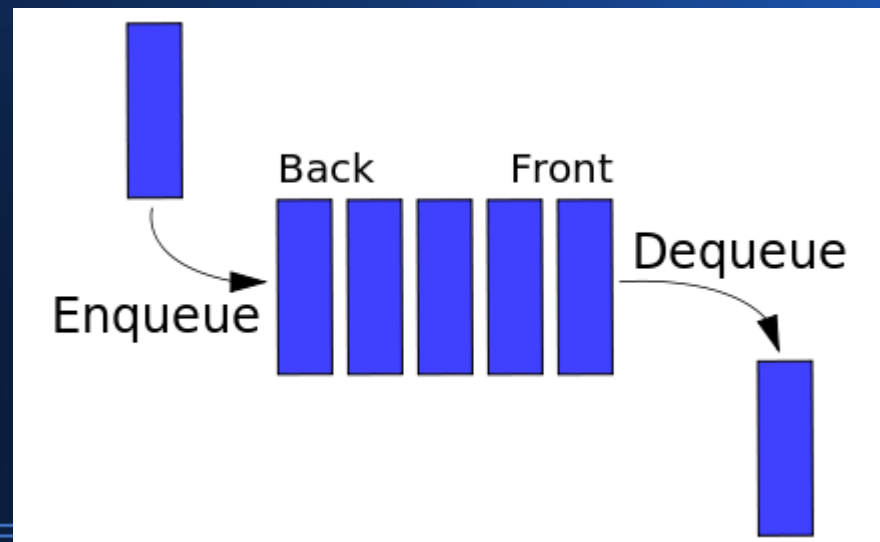
Matrices

- Conjunto de datos organizado en filas y columnas.
- En Python, puede representarse como una lista de listas.

```
In [16]: matriz = [[1,10,100],  
.....:             [2,20,200],  
.....:             [3,30,300]]  
  
In [17]: matriz  
Out[17]: [[1, 10, 100], [2, 20, 200], [3, 30, 300]]  
  
In [18]: type(matriz)  
Out[18]: list  
  
In [19]: len(matriz)  
Out[19]: 3  
  
In [20]:
```

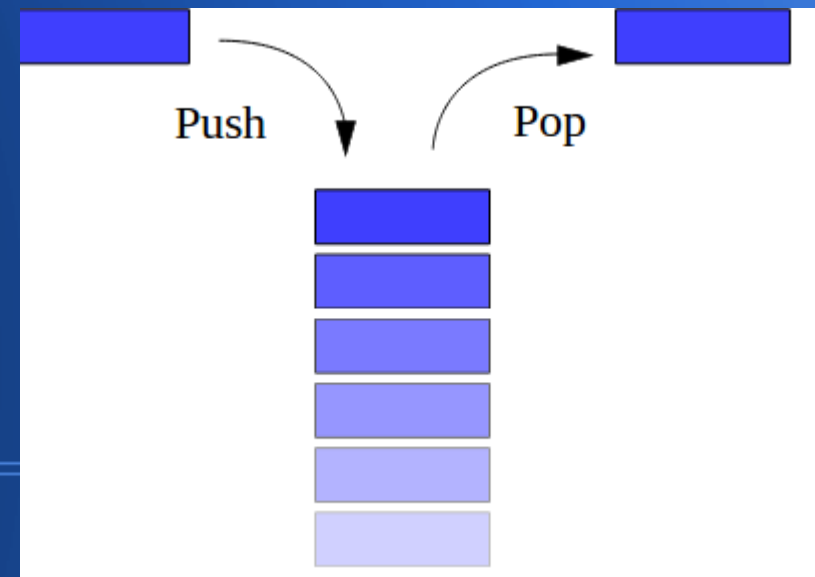
Colas

- Una estructura de datos similar a la lista, en donde el primer elemento en entrar, es el primero en salir.
- Se rigen por el principio FIFO: First In, First Out



Pilas

- Estructura de datos similar a una lista, en donde el último elemento en entrar, es el primero en salir.
- Define 2 operaciones: push y pop.
- Principio LIFO: Last In, First Out.



Diccionarios

- Es una lista de elementos que está indexada con llaves que actúan como palabras clave.
- Se compone de llaves y valores.
- Su implementación hace que las búsquedas sean eficientes.
- Muchos lenguajes no implementan diccionarios, Python si lo hace :-)

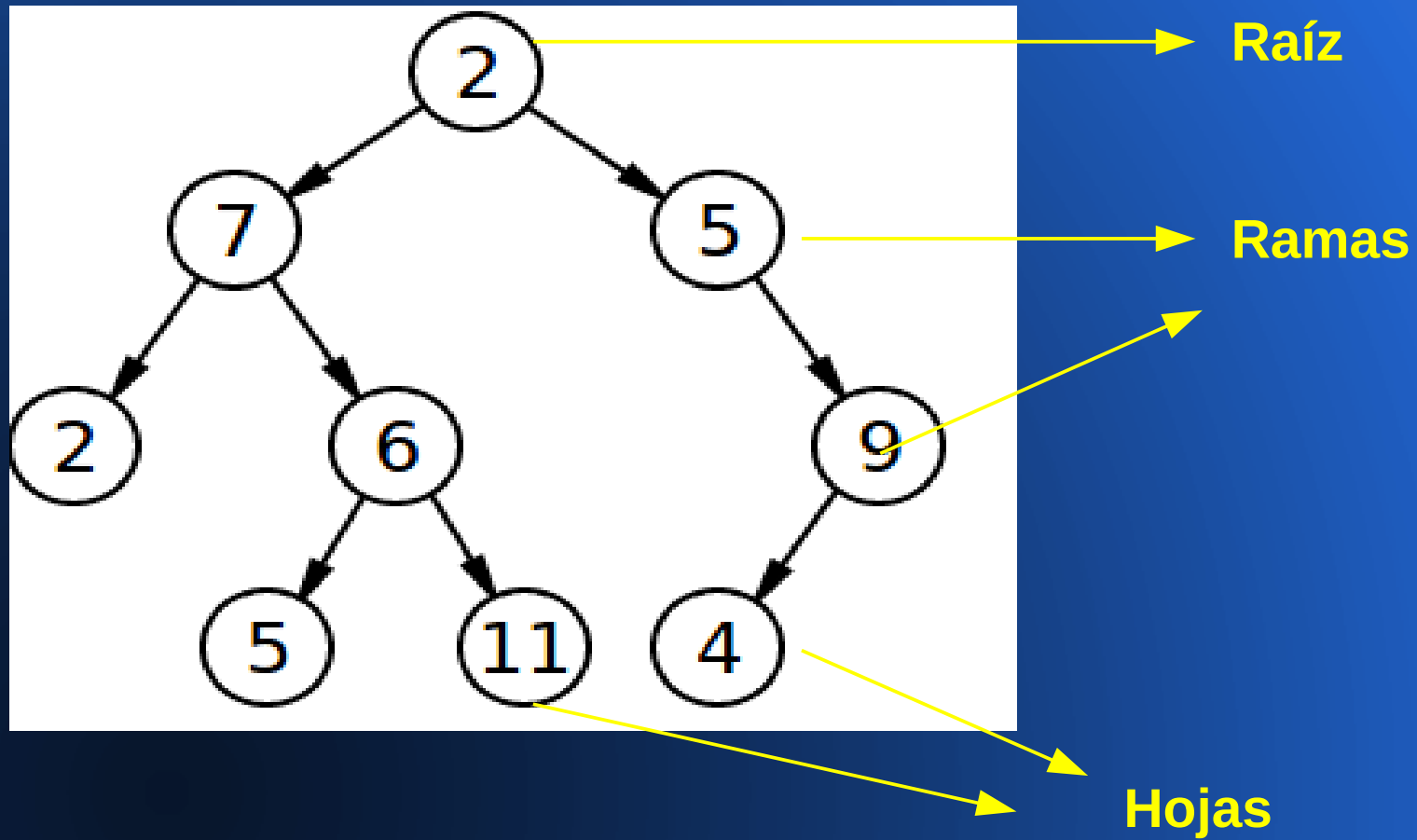
Diccionarios en Python

```
In [56]: cuerpos_celestes = {"sol": "Estrella madre",  
.....:                     "luna" : "Satelite de la Tierra",  
.....:                     "tierra" : "Nuestro hogar",  
.....:                     "marte" : "Planeta Rojo"}  
  
In [57]: type(cuerpos_celestes)  
Out[57]: dict  
  
In [58]: len(cuerpos_celestes)  
Out[58]: 4  
  
In [59]: cuerpos_celestes["tierra"]  
Out[59]: 'Nuestro hogar'  
  
In [60]: cuerpos_celestes["sol"]  
Out[60]: 'Estrella madre'  
  
In [61]: cuerpos_celestes.keys()  
Out[61]: ['luna', 'marte', 'sol', 'tierra']  
  
In [62]: cuerpos_celestes.values()  
Out[62]: ['Satelite de la Tierra', 'Planeta Rojo', 'Estrella madre', 'Nuestro ho  
gar']
```

Árboles

- Estructura de datos compleja compuesta por nodos que presentan jerarquía.
- Inician en un nodo raíz, cada nodo puede tener hijos (o no tenerlos).
- A los nodos intermedios se les conoce como ramas.
- A los nodos sin hijos se les conoce como hojas.
- Python no implementa árboles :-)

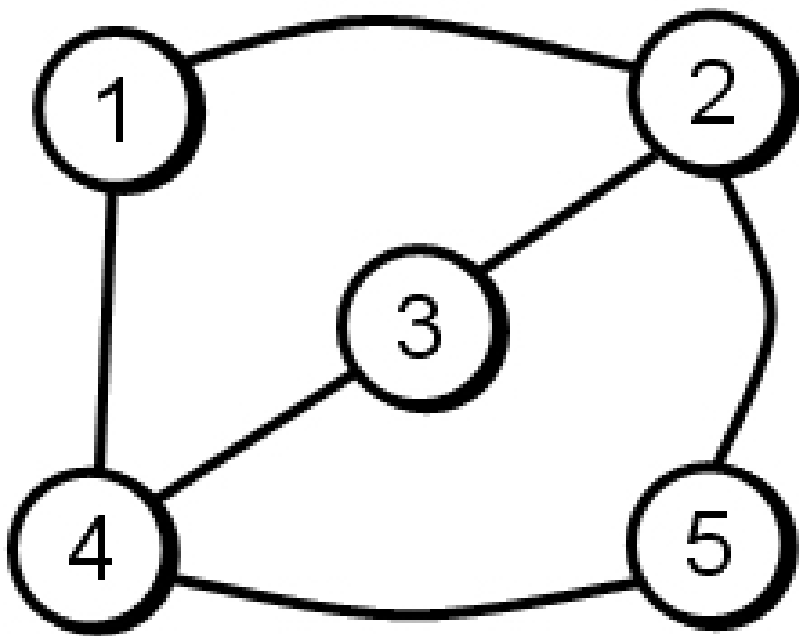
Árboles



Grafos

- Estructura representada por conjunto de vértices (nodos) y aristas que relacionan a los vértices.
- Sirve para representar estructuras donde nos interesen conocer los vecinos (ejemplo: vecinos de partículas).
- Existen algoritmos establecidos para construir y recorrer grafos.
- Python tampoco implementa grafos :'-(

Representación de grafo: matriz de adyacencia



M	1	2	3	4	5
1	0	1	0	1	0
2	1	0	1	0	1
3	0	1	0	1	0
4	1	0	1	0	1
5	0	1	0	1	0