

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación



IIC2115 - Programación como Herramienta para la Ingeniería

Estructuras de datos básicas

Profesor: Hans Löbel

Tipos de dato

- Tipos numéricos

int

enteros

float

racionales

- Tipos textuales

str

texto

- Tipos lógicos (booleanos)

bool

lógico

Operadores

- Aritméticos (para ints y floats principalmente, pueden definirse para otros)



Suma



Resta o
Inverso
aditivo



Producto



División



División
Entera



Resto



Potencia

- Comparación



Menor



Menor
o igual



Mayor



Mayor
o igual



Igual



Distinto

Operadores

- Lógicos (para bools)

not

Negación

and

Y

or

O

A	B	A AND B	A OR B	NOT A
False	False	False	False	True
False	True	False	True	True
True	False	False	True	False
True	True	True	True	False

- Texto (para strs)

+

Concatenación

*

Repetición

Funciones

Nos permiten reutilizar partes de código tantas veces como queramos y generalizar su estructura.



Funciones

- Las funciones no necesariamente retornan un valor, pueden solo realizar cambios o actualizaciones sobre un parámetro (o en el caso de un método, de los atributos de un objeto).
- Los parámetros de entrada tampoco son obligatorios. Se pueden crear funciones que no reciban parámetros.

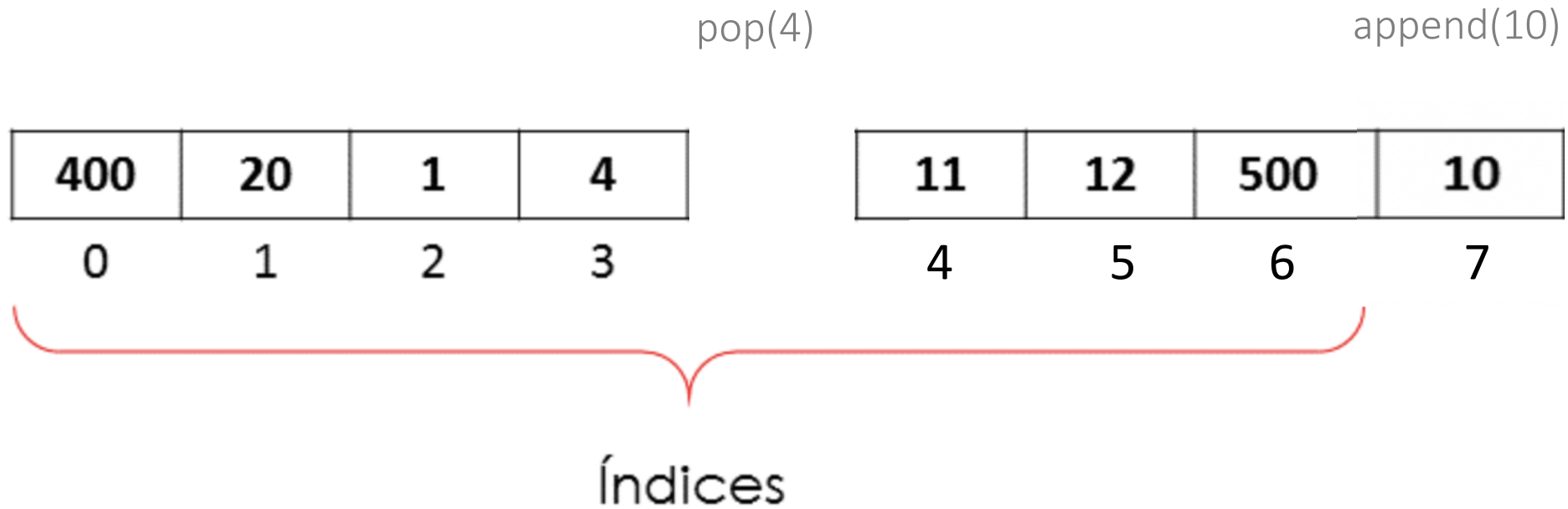
Estructuras de datos

Son un tipo de dato especializado, diseñado para agrupar, almacenar o acceder a la información de manera más eficiente que un tipo de dato básico (como int, float, etc). Las estructuras de datos que veremos en esta clase son las siguientes:

- Listas
- Tuplas
- Diccionarios
- Clases

Listas

- Las listas son estructuras que guardan datos de forma **ordenada**.
- Son mutables (modificables).




Listas en Python

```
l1 = [3,5,63,23]
l2 = list()
l1[1] # 5
l1.append(4) #[3,5,63,23,4]
l1.pop(2) #[3,5,23,4]
```

Tuplas

- Similares a las listas, permiten manejar datos de forma ordenada.
- Al igual que las listas, se accede a los datos mediante índices basados en el orden que fueron ingresados.
- A diferencia de las listas, son **inmutables**.

400	20	1	4	10	11	12	500
0	1	2	3	4	5	6	7



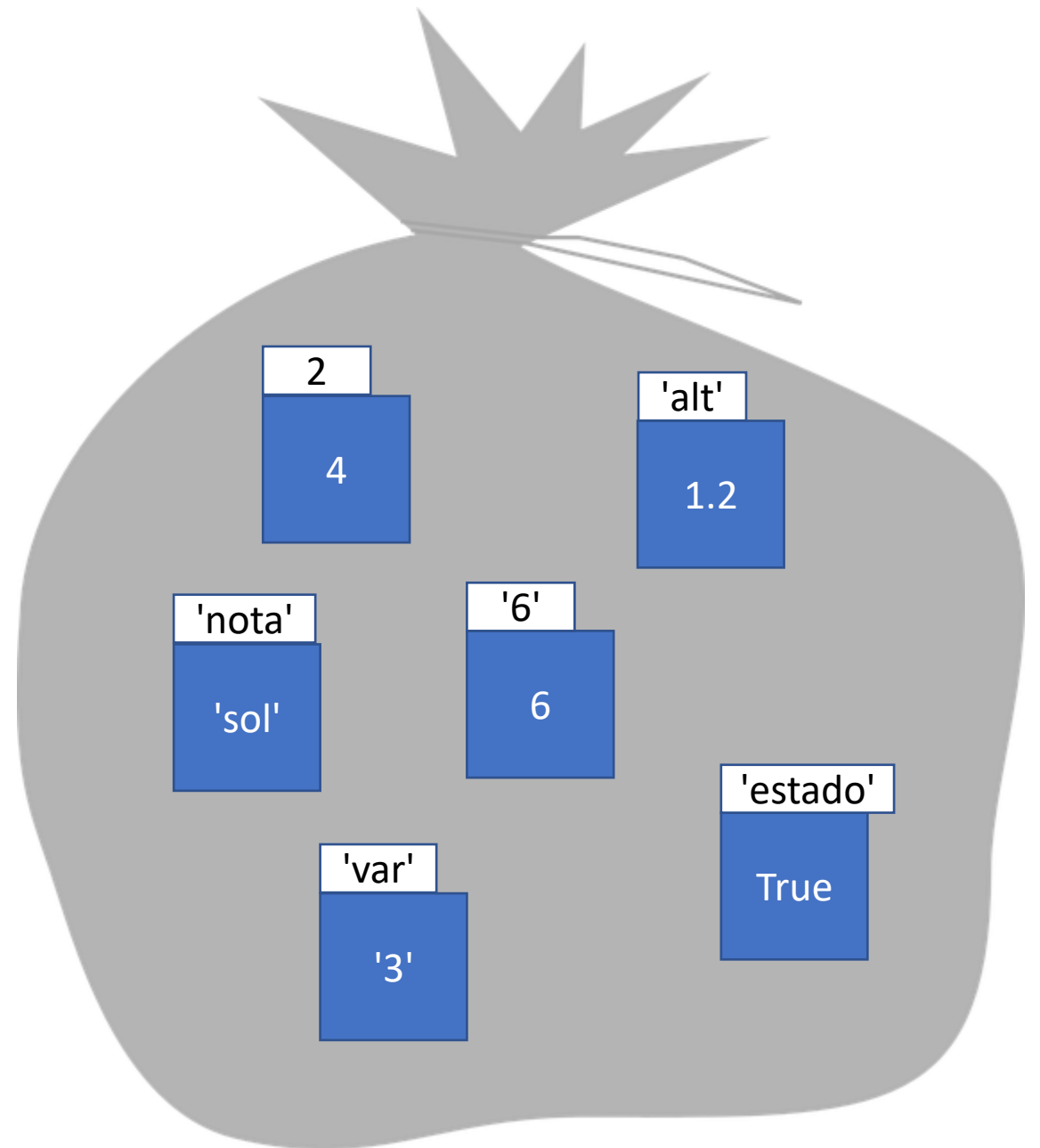
Índices

Tuplas en Python

```
t1 = (3,5,63,23)
t2 = tuple()
t1[1] # 5
t1.append(4) #error
t1.pop(2) #error
```

Diccionarios

- Permiten almacenar datos basados en una asociación de pares de elementos, a través de una relación **llave-valor**.
- Acceso a valores a través de la llave es instantáneo, no se necesita realizar una búsqueda (análogo a un índice).
- Se prefiere a una lista cuando el caso de uso más común no implica revisar todos los elementos, sino solo algunos fácilmente encontrables a través de la llave.



Diccionarios en Python

```
d1 = {'nom':'Jen', 'edad':63}  
d2 = dict()  
d1['nom'] # 'Jen'  
d1.update('año': 1960) #error
```

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación



IIC2115 - Programación como Herramienta para la Ingeniería

Estructuras de datos básicas

Profesor: Hans Löbel