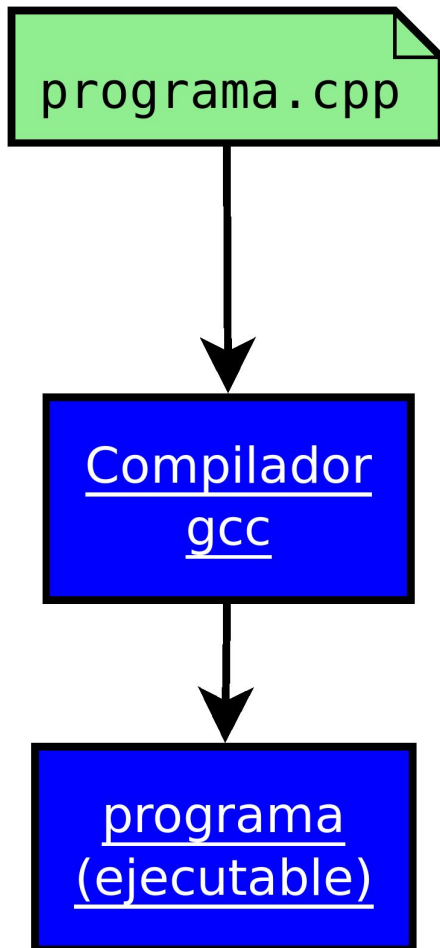


# Introducción a la Computación

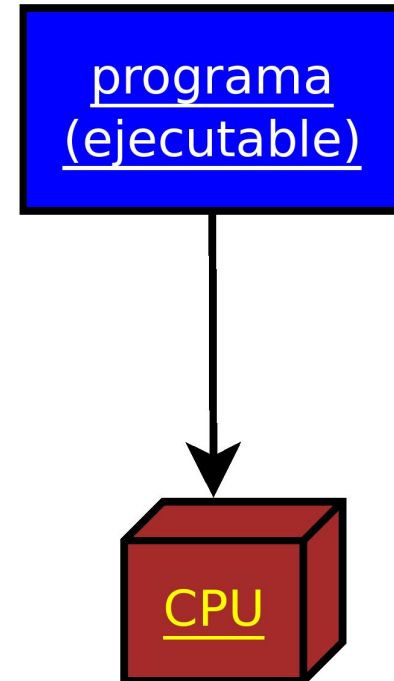
Introducción a Python

# Lenguajes compilados

Etapa de compilación

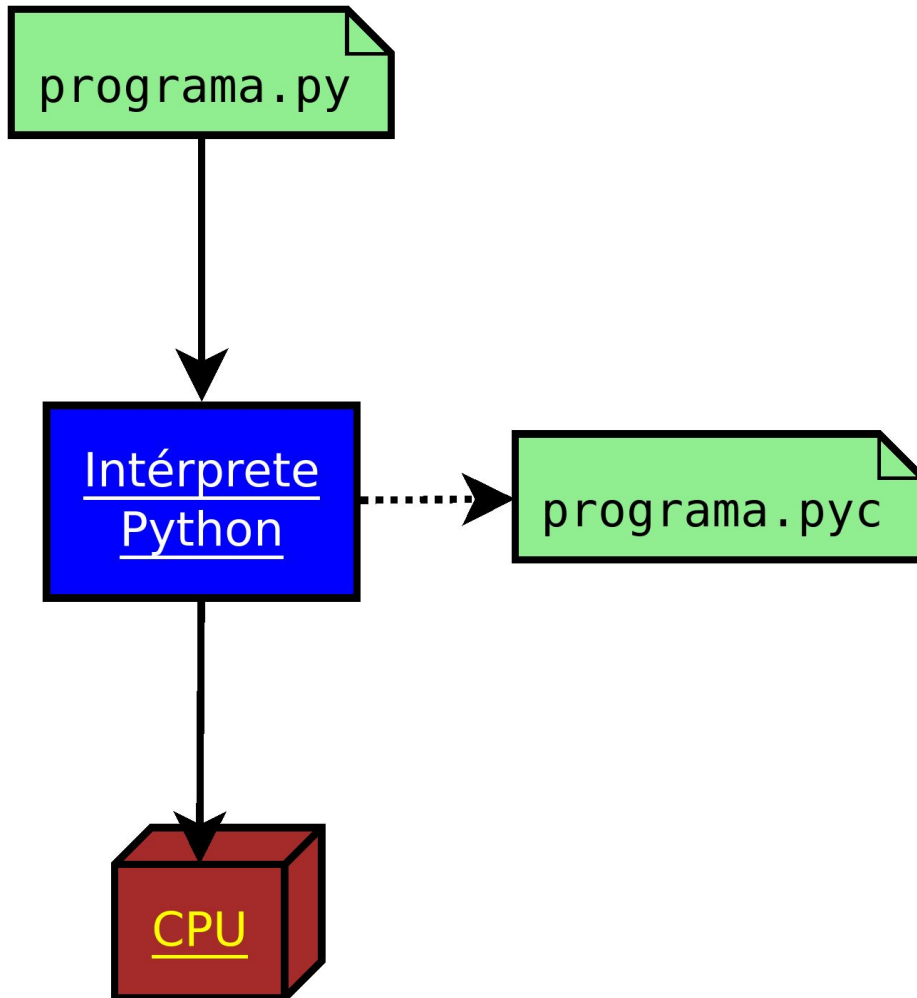


Etapa de ejecución

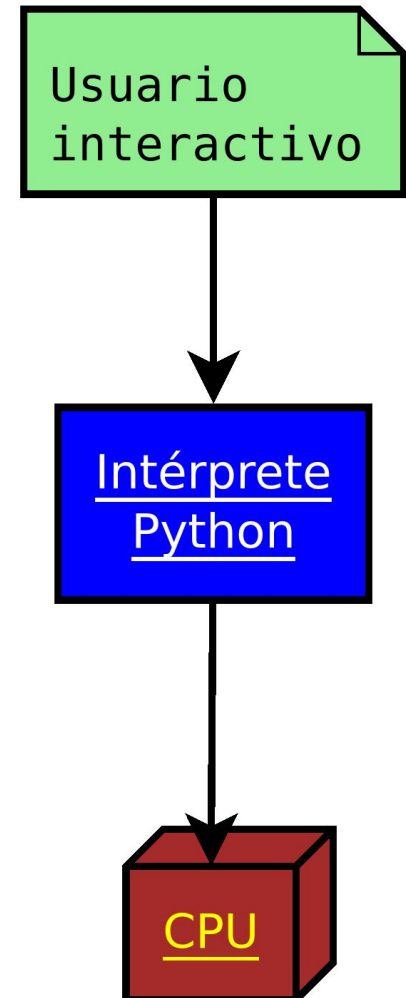


# Lenguajes interpretados (Python)

## Etapa de “Compilación”



## Sesión interactiva



# Python

## Intérprete interactivo

```
$ python
```

```
Python 2.7 (#1, Feb 28 2010, 00:02:06)
```

```
Type "help", "copyright", "credits" or "license" for more  
information.
```

```
>>>
```

## Evaluador de expresiones

```
>>> 2+2
```

```
4
```

```
>>> 2+2.0
```

```
4.0
```

```
>>> "hola"
```

```
'hola'
```

```
>>> 'hola'
```

```
'hola'
```

```
>>> "Hola," + " mundo"
```

```
'Hola, mundo'
```

# Python

## Interprete interactivo

```
$ python
Python 2.7 (#1, Feb 28 2010, 00:02:06)
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

## Evaluador de expresiones

```
>>> 2+2
4
>>> 2+2.0
4.0
>>> "hola"
'hola'
>>> 'hola'
'hola'
>>> "Hola," + " mundo"
'Hola, mundo'
```

# Python

## Interprete interactivo

```
$ python
Python 2.7 (#1, Feb 28 2010, 00:02:06)
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

## Evaluador de expresiones

```
>>> 2+2
4
>>> 2+2.0
4.0
>>> "hola"
'hola'
>>> 'hola'
'hola'
>>> "Hola," + " mundo"
'Hola, mundo'
```

# Python

## Interprete interactivo

```
$ python
Python 2.7 (#1, Feb 28 2010, 00:02:06)
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

## Evaluador de expresiones

```
>>> 2+2
4
>>> 2+2.0
4.0
>>> "hola"
'hola'
>>> 'hola'
'hola'
>>> "Hola," + " mundo"
'Hola, mundo'
```

# Python

## Interprete interactivo

```
$ python
Python 2.7 (#1, Feb 28 2010, 00:02:06)
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

## Evaluador de expresiones

```
>>> 2+2
4
>>> 2+2.0
4.0
>>> "hola"
'hola'
>>> 'hola'
'hola'
>>> "Hola," + " mundo"
'Hola, mundo'
```



# Python

## Caracteres de escape

```
>>> 'doesn't'  
File "<stdin>", line 1  
    'doesn't'  
      ^  
SyntaxError: invalid syntax
```

```
>>> 'doesn\'t'  
"doesn't"
```

```
>>> "doesn't"  
"doesn't"
```

# Python

## Caracteres de escape

```
>>> 'doesn't'  
File "<stdin>", line 1  
    'doesn't'  
      ^  
SyntaxError: invalid syntax
```

```
>>> 'doesn\'t'  
"doesn't"
```

```
>>> "doesn't"  
"doesn't"
```

# Python

## Caracteres de escape

```
>>> 'doesn't'  
File "<stdin>", line 1  
    'doesn't'  
      ^  
SyntaxError: invalid syntax
```

```
>>> 'doesn\'t'  
"doesn't"
```

```
>>> "doesn't"  
"doesn't"
```

# Python

## Variables

```
>>> base = 20
>>> altura = 5*9
>>> base * altura
900

>>> x = " cinco "
>>> '<' + x*5 + '>'
'< cinco  cinco  cinco  cinco  cinco >'

>>> x = "Hola"
>>> x[3]
'a'
>>> x[1:3]
'ol'

>>> s = 'supercalifragilisticoespialidoso'
>>> len(s)
32
```

# Python

## Variables

```
>>> base = 20
>>> altura = 5*9
>>> base * altura
900

>>> x = " cinco "
>>> '<' + x*5 + '>'
'< cinco cinco cinco cinco cinco >'

>>> x = "Hola"
>>> x[3]
'a'
>>> x[1:3]
'ol'

>>> s = 'supercalifragilisticoespialidoso'
>>> len(s)
32
```

# Python

## Variables

```
>>> base = 20
>>> altura = 5*9
>>> base * altura
900

>>> x = " cinco "
>>> '<' + x*5 + '>'
'< cinco cinco cinco cinco cinco >'

>>> x = "Hola"
>>> x[3]
'a'
>>> x[1:3]
'ol'

>>> s = 'supercalifragilisticoespialidoso'
>>> len(s)
32
```

# Python

## Variables

```
>>> base = 20
>>> altura = 5*9
>>> base * altura
900

>>> x = " cinco "
>>> '<' + x*5 + '>'
'< cinco  cinco  cinco  cinco  cinco >'

>>> x = "Hola"
>>> x[3]
'a'
>>> x[1:3]
'ol'

>>> s = 'supercalifragilisticoespialidoso'
>>> len(s)
32
```

# Python

## Asignación simultánea

```
>>> x = y = 0
>>> x
0
>>> y
0

>>> x, y = 3, 4
>>> x
3
>>> y
4
```

## Tipado dinámico

```
>>> x = 45
>>> x
45
>>> x = "Pirulo"
>>> x
'Pirulo'
```



# Python

## Asignación simultánea

```
>>> x = y = 0
>>> x
0
>>> y
0

>>> x, y = 3, 4
>>> x
3
>>> y
4
```

## Tipado dinámico

```
>>> x = 45
>>> x
45
>>> x = "Pirulo"
>>> x
'Pirulo'
```

# Python

## Asignación simultánea

```
>>> x = y = 0
>>> x
0
>>> y
0

>>> x, y = 3, 4
>>> x
3
>>> y
4
```

## Tipado dinámico

```
>>> x = 45
>>> x
45
>>> x = "Pirulo"
>>> x
'Pirulo'
```

# Python

## Listas

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> a
['spam', 'huevos', 100, 1234]

>>> a[0]
'spam'

>>> a[3]
1234

>>> a[-2]
100

>>> a[1:3]
['huevos', 100]

>>> a[:2]
['spam', 'huevos']

>>> a[:2] + ['panceta', 2*2]
['spam', 'huevos', 'panceta', 4]
```

# Python

## Listas

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> a
['spam', 'huevos', 100, 1234]

>>> a[0]
'spam'

>>> a[3]
1234

>>> a[-2]
100

>>> a[1:3]
['huevos', 100]

>>> a[:2]
['spam', 'huevos']

>>> a[:2] + ['panceta', 2*2]
['spam', 'huevos', 'panceta', 4]
```

# Python

## Listas

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> a
['spam', 'huevos', 100, 1234]

>>> a[0]
'spam'

>>> a[3]
1234

>>> a[-2]
100

>>> a[1:3]
['huevos', 100]

>>> a[:2]
['spam', 'huevos']

>>> a[:2] + ['panceta', 2*2]
['spam', 'huevos', 'panceta', 4]
```

# Python

## Listas

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> a
['spam', 'huevos', 100, 1234]

>>> a[0]
'spam'

>>> a[3]
1234

>>> a[-2]
100

>>> a[1:3]
['huevos', 100]

>>> a[:2]
['spam', 'huevos']

>>> a[:2] + ['panceta', 2*2]
['spam', 'huevos', 'panceta', 4]
```

# Python

## Listas por referencia

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> b = a
>>> b[2] = "perejil"
>>> a
['spam', 'huevos', 'perejil', 1234]
```

## Otros ejemplos

```
>>> 2*a[1:3] + ['tomate', 'lechuga']
['huevos', 'perejil', 'huevos', 'perejil', 'tomate', 'lechuga']

>>> a = [2, "pirulo", [2, "montoto"]]
>>> a
[2, 'pirulo', [2, 'montoto']]
```

# Python

## Listas por referencia

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> b = a
>>> b[2] = "perejil"
>>> a
['spam', 'huevos', 'perejil', 1234]
```

## Otros ejemplos

```
>>> 2*a[1:3] + ['tomate', 'lechuga']
['huevos', 'perejil', 'huevos', 'perejil', 'tomate', 'lechuga']

>>> a = [2, "pirulo", [2, "montoto"]]
>>> a
[2, 'pirulo', [2, 'montoto']]
```



# Python

## Listas por referencia

```
>>> a = ['spam', 'huevos', 100, 1234]
>>> b = a
>>> b[2] = "perejil"
>>> a
['spam', 'huevos', 'perejil', 1234]
```

## Otros ejemplos

```
>>> 2*a[1:3] + ['tomate', 'lechuga']
['huevos', 'perejil', 'huevos', 'perejil', 'tomate', 'lechuga']

>>> a = [2, "pirulo", [2, "montoto"]]
>>> a
[2, 'pirulo', [2, 'montoto']]
```

# Python

## Estructuras de control: condicionales

```
>>> x = 5
>>> if x < 0:
...     print 'negativo'
... elif x == 0:
...     print 'cero'
... else:
...     print 'positivo'
...
positivo
```

# Python

## Estructuras de control: bloques

```
>>> x = 5
>>> if x < 0:
... [TAB]print 'negativo'
... elif x == 0:
...     print 'cero'
... else:
...     print 'positivo'
...
positivo
```

# Python

## Estructuras de control: ciclos

```
>>> a, b = 0, 1
>>> while b < 10:
...     print b
...     a, b = b, a+b
...
1
1
2
3
5
8
```

```
>>> a, b = 0, 1
>>> while b < 10:
...     print b,
...     a, b = b, a+b
...
1 1 2 3 5 8
```

# Python

## Nueva línea

```
>>> # Serie de fibonacci
... a, b = 0, 1
>>> while b < 10:
...     print b
...     a, b = b, a+b
...
1
1
2
3
5
8
```

```
>>> a, b = 0, 1
>>> while b < 10:
...     print b,
...     a, b = b, a+b
...
1 1 2 3 5 8
```

# Python

## Archivos .py

```
$ cat fib.py
# Serie de fibonacci
a, b = 0, 1
while b < 10:
    print b
    a, b = b, a+b
```

```
$ python fib.py
1
1
2
3
5
8
```

# Python

## Archivos .py: comentarios

```
$ cat fib.py
# Serie de fibonacci
a, b = 0, 1
while b < 10:
    print b
    a, b = b, a+b

$ python fib.py
1
1
2
3
5
8
```

# Python

## Archivos .py

```
$ cat fib.py
a, b = 0, 1
while b < 10:
    print b
    a, b = b, a+b
```

```
$ python fib.py
```

```
1
1
2
3
5
8
```



# Python

## Archivos .py "ejecutables"

```
$ chmod +x fib2.py
$ cat fib2.py
#!/usr/bin/python
```

```
a, b = 0, 1
while b < 10:
    print b
    a, b = b, a+b
```

```
$ fib2.py
1
1
2
3
5
8
```

# Python

## Archivos .py ejecutables

```
$ cat fib2.py
#!/usr/bin/python
```

```
a, b = 0, 1
while b < 10:
    print b
    a, b = b, a+b
```

```
$ ./fib2.py
```

```
1
1
2
3
5
8
```

# Python

## Argumentos de programa

```
$ cat arg.py
#!/usr/bin/python

import sys

print sys.argv

$ ./arg.py hola 1 3 primo
['./arg.py', 'hola', '1', '3', 'primo']
```

# Python

## Argumentos de programa

```
$ cat arg.py
#!/usr/bin/python

import sys

print sys.argv

$ ./arg.py hola 1 3 primo
['./arg.py', 'hola', '1', '3', 'primo']
```

# Python

## Argumentos de programa

```
$ cat arg.py
#!/usr/bin/python

import sys

print sys.argv

$ ./arg.py hola 1 3 primo
['./arg.py', 'hola', '1', '3', 'primo']
```

# Ejercicios!

Resolver nuevamente las guías 1 y 2 usando...



# Python

