


Licenciatura en Ciencias de la Computación



Algoritmos y Estructuras de Datos I




UNIVERSIDAD
NACIONAL DE CUYO



FACULTAD DE INGENIERIA
en acción continua...

Dr. Carlos A. Catania
Ing. Lucía Cortés
Mg. Lic. Javier Rosenstein



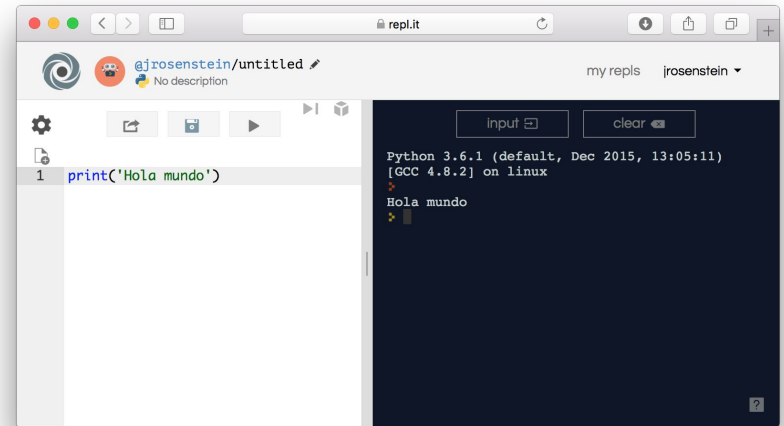
Pseudo-Python

- Sintaxis Elegante - Sin delimitadores de línea ni bloque
- Es un lenguaje Interpretado
- Si bien permite el uso de librerías varias

no usaremos **NINGUNA** librería de python salvo las así indicadas por la catedra como ser algo1.py

● Prácticas de algoritmos

1. Para realizar las prácticas vamos a usar el entorno **repl.it**
2. Para poder ejecutar código python entonces vamos a abrir la herramienta online y ejecutaremos el código que deseemos probar...
3. Es conveniente crearnos una cuenta y loguearnos para guardar nuestras prácticas en la misma.





Tipos de datos básicos

Enteros -

ej: 245- 0

Reales - Decimales

ej: 23.45 - 2.0

chr - caracter

ej: 'a'

bool - Lógicos True/False

(cualquier valor distinto de 0 es True)



Tipos de datos básicos II

Cadena de caracteres:

La estructura **String** se utiliza para representar una cadena de caracteres:

```
1. S=String("hola mundo")  
2. print(S[1])  
   'o'
```



condicionales

Pseudocódigo

```
si (condición) hacer  
    sentencias 1  
  
sino  
    sentencias 2  
  
fin_si
```

Python

```
if x < 0:  
    x = 0  
    print('negativo')  
elif x == 0:  
    print('Cero')  
elif x == 1:  
    print('Simple')  
else:  
    print('Mas')
```



Estructuras de Control - for

Pseudocódigo

desde variable = 2 hasta 4 hacer

...

sentencias

...

fin_desde

Python

```
for n in range(2, 5):  
    print('valor:',n)
```

```
valor: 2  
valor: 3  
valor: 4
```

```
for n in range(2, 10, 3):  
    print('valor:',n)
```

```
valor: 2  
valor: 5  
valor: 8
```



Estructuras de Control - while

Pseudocódigo

```
n=0  
  
mientras n<8 hacer  
    escribir(...)  
  
    n=n+2  
  
fin_mientras
```

Python

```
n=0  
while (n<8):  
    print('valor:',n)  
    n=n+2  
  
valor: 0  
valor: 2  
valor: 4  
valor: 6
```




Funciones

```
def muestraLista(n):  
    while (n>0):  
        print('valor:',n)  
        n=n-1  
    return  
muestraLista(5)
```

```
valor: 5  
valor: 4  
valor: 3  
valor: 2  
valor: 1
```

importar librerías

```
from algo1 import *           #importamos la librería de la materia algo1.py
```



Ingresar valores por teclado:

```
valor_entero = input_int("Ingrese la cantidad de filas:")  
con_decimales = input_real("Ingrese importe con decimales:")  
nombre = input_str("Ingrese el texto alfanumérico:")
```

Funciones matemáticas:

```
import math                   #importamos la librería de funciones matemáticas de repl.it  
raiz = math.sqrt(16)  
valor=math.pow(2,3)          #elevo 2 al cúbo...
```

uso de arreglos en repl.it

```
from algo1 import *
```



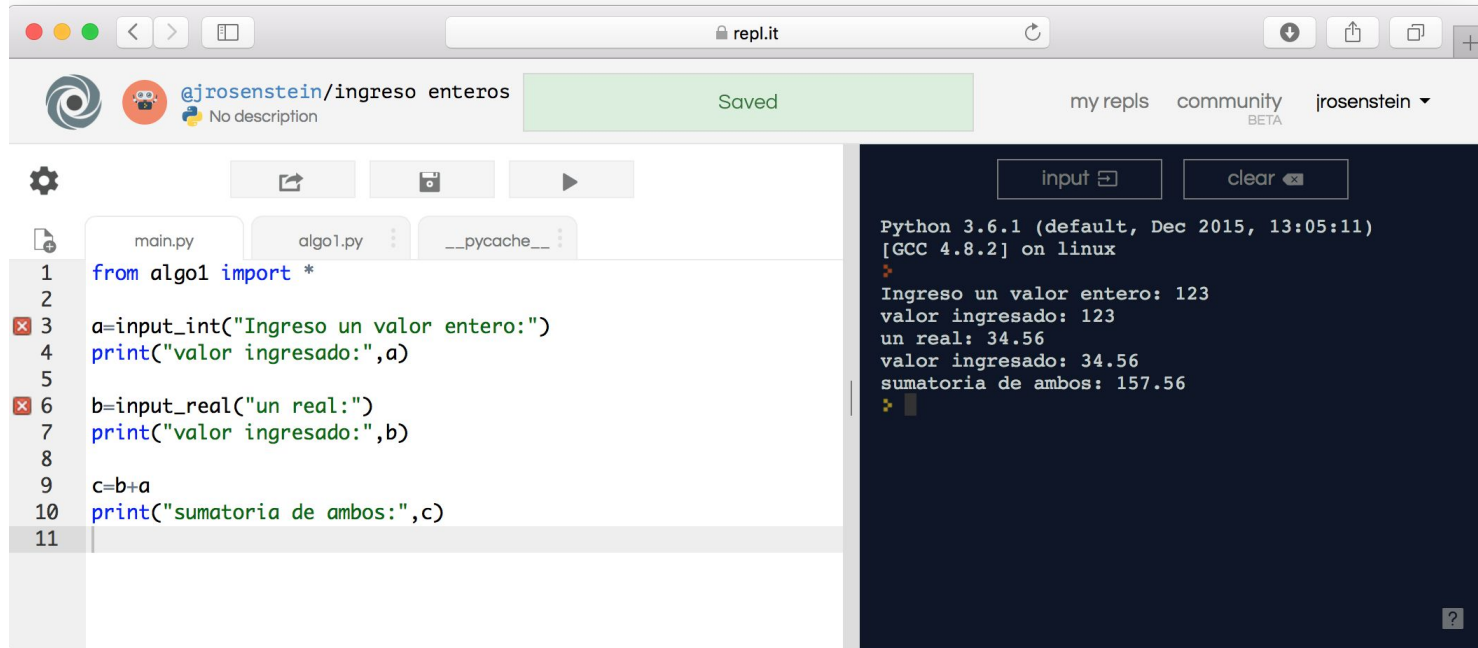
```
vector_a=Array(10,0)      #Declaramos un arreglo de longitud 10 de tipo entero
vector_b=Array(10,0.0)    #Declaramos un arreglo de longitud 10 de tipo float
vector_c=Array(10,Array(10,'c')) #Declaramos un arreglo matriz de 10x10 de tipo carácter

vector_a[1]=45            #
vector_c[1][4]='f'
print(vector_a[1])        #que hay en la celda 1?
print(vector_c[1][1])    #que hay en la celda 1 1?
len(vector_a)             #cual es la longitud del arreglo?
print(vector_a)           #cual es el contenido de todo el arreglo?

a[1]=a[1]+5               #sumamos 5 a la celda 1"
```

Ejercicios en repl.it

Ingreso de valores enteros y reales...



The screenshot shows a web browser window with the URL `repl.it`. The page header includes the user profile `@jrosenstein/ingreso enteros`, a `Saved` status, and navigation links for `my repls`, `community`, and the user `jrosenstein`. The main area is divided into a code editor on the left and a terminal on the right.

The code editor displays a Python script in `main.py` with the following content:

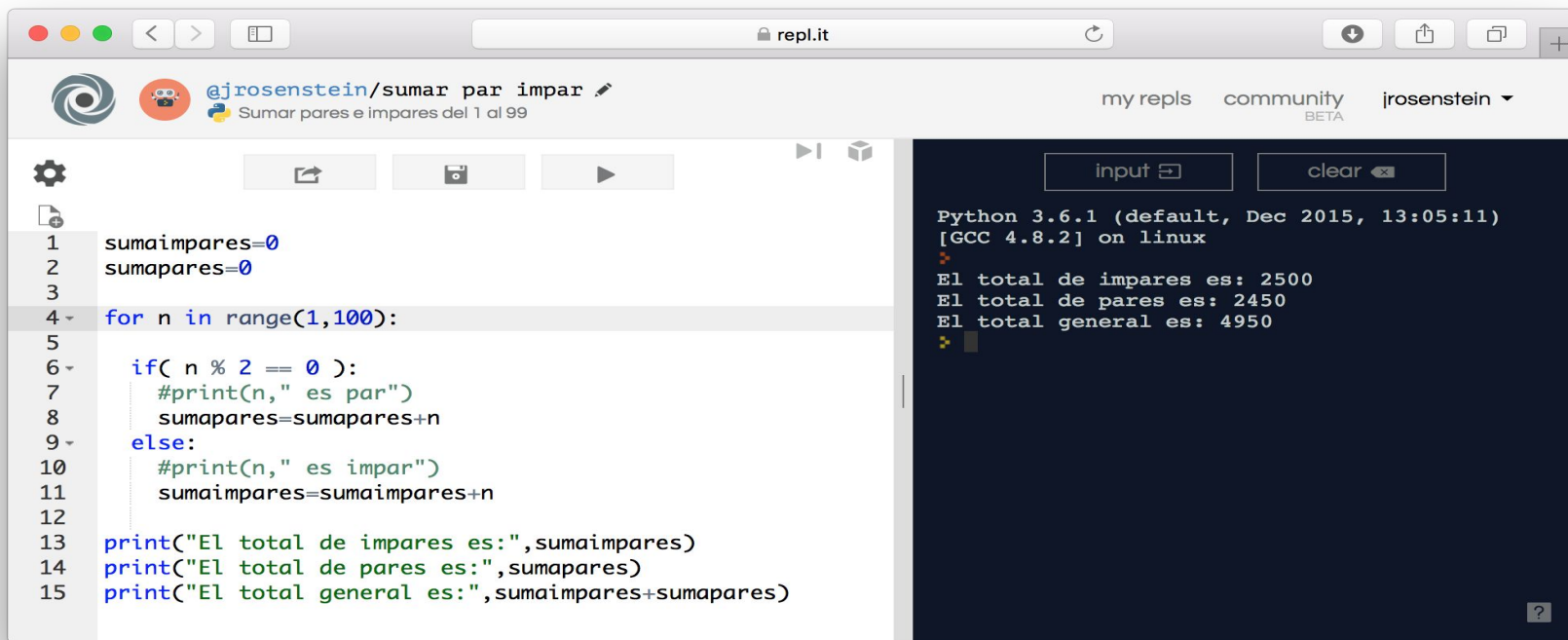
```
1 from algo1 import *
2
3 a=input_int("Ingreso un valor entero:")
4 print("valor ingresado:",a)
5
6 b=input_real("un real:")
7 print("valor ingresado:",b)
8
9 c=b+a
10 print("sumatoria de ambos:",c)
11
```

The terminal on the right shows the output of the program after running:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
Ingreso un valor entero: 123
valor ingresado: 123
un real: 34.56
valor ingresado: 34.56
sumatoria de ambos: 157.56
```

Ejercicios en repl.it

Escriba un programa que sea capaz de sumar los números impares del 1 al 99 inclusive.



The screenshot shows a web browser window with the URL `repl.it`. The page title is `@jrosenstein/sumar par impar` with a subtitle `Sumar pares e impares del 1 al 99`. The user `jrosenstein` is logged in. The code editor on the left contains the following Python code:

```
1 sumaimpares=0
2 sumapares=0
3
4 for n in range(1,100):
5
6     if( n % 2 == 0 ):
7         #print(n, " es par")
8         sumapares=sumapares+n
9     else:
10        #print(n, " es impar")
11        sumaimpares=sumaimpares+n
12
13 print("El total de impares es:",sumaimpares)
14 print("El total de pares es:",sumapares)
15 print("El total general es:",sumaimpares+sumapares)
```

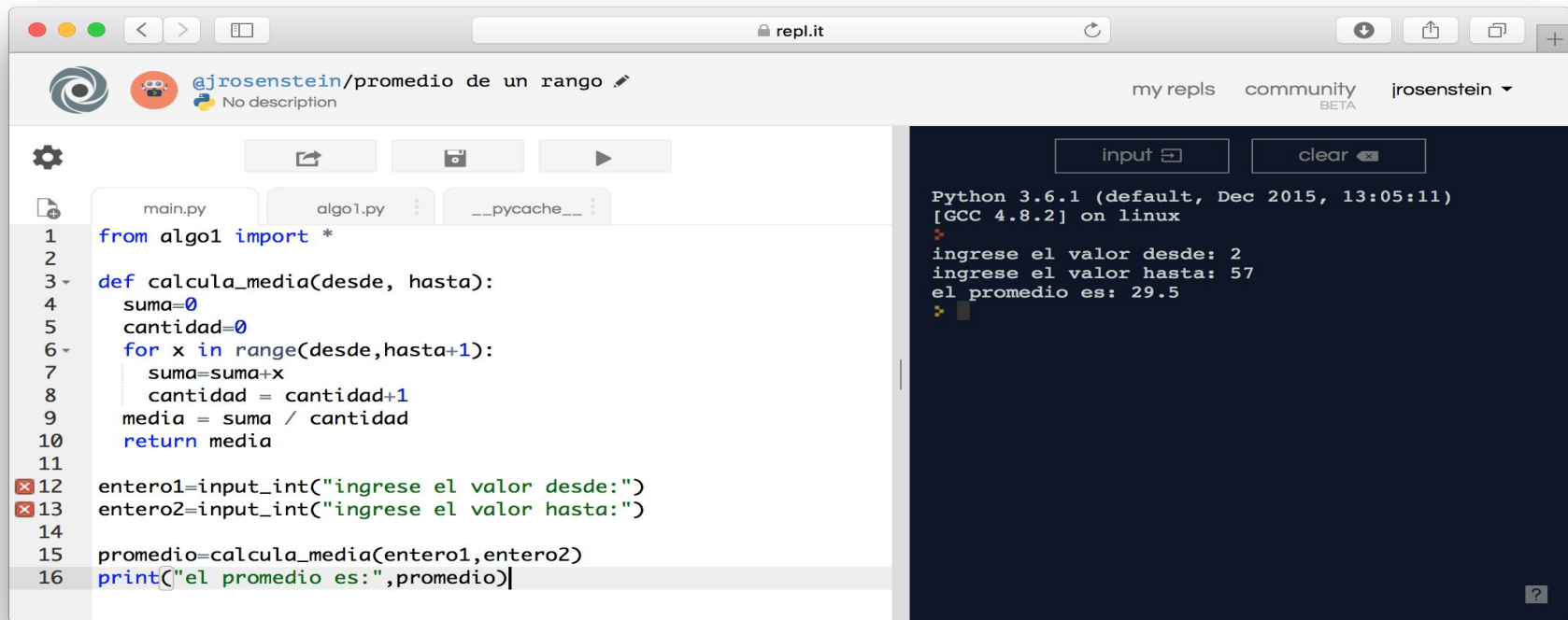
The output terminal on the right shows the execution results:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux

El total de impares es: 2500
El total de pares es: 2450
El total general es: 4950
```

Ejercicios en repl.it

Escriba un programa que permita ingresar los límites enteros de un rango de valores y calcule el promedio de los mismos mediante una función.



The screenshot shows a web browser window with the URL `repl.it`. The page title is `@jrosenstein/promedio de un rango`. The interface includes a file explorer on the left with files `main.py`, `algo1.py`, and `__pycache__`. The main editor displays a Python script in `main.py` with the following code:

```
1 from algo1 import *
2
3 def calcula_media(desde, hasta):
4     suma=0
5     cantidad=0
6     for x in range(desde,hasta+1):
7         suma=suma+x
8         cantidad = cantidad+1
9     media = suma / cantidad
10    return media
11
12 entero1=input_int("ingrese el valor desde:")
13 entero2=input_int("ingrese el valor hasta:")
14
15 promedio=calcula_media(entero1,entero2)
16 print("el promedio es:",promedio)
```

On the right, the terminal output shows the execution of the program:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
ingrese el valor desde: 2
ingrese el valor hasta: 57
el promedio es: 29.5
```

Ejercicios en repl.it



- Escriba un programa que permita ingresar 2 valores enteros y realice la siguiente operación:

-Mediante el uso de una función, mostrar un listado en orden decreciente desde el valor de la suma de ambos hasta 0, teniendo en cuenta que si la suma es mayor a 50 entonces hacerlo de 5 en 5, si no de 2 en 2.

- Escriba un programa que permita ingresar 3 valores reales y realice la siguiente operación:
 - Comparar los valores y mostrarlos ordenados de menor a mayor.
 - calcular y mostrar el promedio de los 2 valores extremos.