

# Ayudantía de Programación

A dark blue background filled with various white and light blue icons representing technology and computing. These include a server rack, a camera, a CD, a smartphone, a laptop, a game controller, a wrench and screwdriver, a fan, a USB drive, a floppy disk, a hard drive, a mouse, a keyboard, a printer, a magnifying glass, a circuit board, and a power plug. The icons are scattered across the slide, creating a tech-themed backdrop.

## Estructuras Condicionales Funciones Strings

Gonzalo Fernández

IWI-131 01-10-2018

# Temas de la Ayudantía de Hoy

- Conciciones: if, elif, else
- While
- Funciones
- Strings
- Ejercicios de Certámenes

# Dudas

# If – Elif – Else

```
if x == 5:  
    print x, "es igual a 5"  
elif x == 6:  
    print x, "es igual a 6"  
else:  
    print x, "no es ni 5 ni 6"
```

# If – Elif – Else

```
if x == 5:  
    print x, "es igual a 5"  
else:  
    print x, "no es 5"
```

# If – Elif – Else

```
if x == 5:  
    print x, "es igual a 5"  
elif x == 6:  
    print x, "es igual a 6"
```

# Par o Impar

Realice un programa que solicite un número al usuario y muestre en pantalla si este es par o impar

Ingrese un numero: 5  
El numero ingresado es par

# Par o Impar

```
x = int(raw_input("Ingrese un numero: "))  
if x % 2 == 0:  
    print "El numero ingresado es par"  
elif x % 2 != 0:  
    print "El numero ingresado es impar"
```



# Par o Impar

```
x = int(raw_input("Ingrese un numero: "))  
if x % 2 == 0:  
    print "El numero ingresado es par"  
else:  
    print "El numero ingresado es impar"
```

# While

```
x = 0  
while x < 5:  
    x += 1  
    print x
```

# While

```
x = 0
```

```
x = int(raw_input("Ingrese un numero mayor a 5: "))
```

```
while x < 5:
```

```
    x = int(raw_input("Ingrese un numero mayor a 5: "))
```

```
print "Gracias por ingresar un numero mayor a 5"
```

```
print "El numero ingresado es: ", x
```

# Conjetura de Collatz

Realice un programa que solicite un número al usuario y muestre en pantalla el desarrollo de la Conjetura de Collatz hasta que sea 1

Formalmente, esto equivale a una función  $f : \mathbb{N} \mapsto \mathbb{N}$ :

$$f(n) = \begin{cases} \frac{n}{2}, & \text{si } n \text{ es par} \\ 3n + 1, & \text{si } n \text{ es impar} \end{cases}$$

# Conjetura de Collatz

```
x = int(raw_input("Ingrese un numero: "))

while x != 1:
    if x % 2 == 0:
        x = x/2
    else:
        x = 3*x + 1
    print x
```

# Conjetura de Collatz

```
x = int(raw_input("Ingresa un numero: "))

while x != 1:
    if x % 2 == 0:
        x = x/2
    elif x % 2 != 0:
        x = 3*x + 1
    print x
```

# Conjetura de Collatz

```
x = int(raw_input("Ingresa un numero: "))

while x != 1:
    if x % 2 == 0:
        x = x/2
    elif x % 2 != 0:
        x = 3*x + 1
    print x
```

# Conjetura de Collatz

```
1 x = int(raw_input("Ingrese un numero: "))
2
3 ▼ while x != 1:
4     if x % 2 == 0:
5         x = x/2
6     else:
7         x = 3*x + 1
8     print x
9
```

```
Ingrese un numero: 24
12
6
3
10
5
16
8
4
2
1
```



# Funciones

```
def hola_mundo():  
    print "Hola Mundo"
```

```
hola_mundo()
```

# Funciones

```
def par(x):  
    if x % 2 == 0:  
        return "par"  
    else:  
        return "impar"
```

```
y = int(raw_input("Ingrese un numero: "))  
print y, "es un numero", par(y)
```

# Funciones

```
def p(x, y):  
    n = 0  
    r = 1  
  
    while n < y:  
        r *= x  
        n += 1  
  
    return r  
  
print p(5, 9)
```

```
def too(y):
    if foo(y):
        y = y / 2
    else:
        y = y + 3
    return y
```

```
while (t):
    w = too(w)
    d = too(d)
    if w-1 == d:
        t = False
print w, d
```

[illegible]

# Strings

```
a = "hola"
```

```
print a[0] # h
```

```
print a[1] # o
```

```
print a[2] # l
```

```
print a[3] # a
```

```
print a[-1] # a
```

```
print a[-2] # l
```

```
print a[-3] # h
```

```
print a[-4] # o
```

# Strings

```
def contar_letras(palabra, letra):  
    n = 0  
    r = 0  
    while n < len(palabra):  
        if palabra[n] == letra:  
            r += 1  
        n += 1  
    return r
```

```
a = raw_input("Ingresa algo: ")  
print a, "tiene", contar_letras(a, "b"), "letras b"
```

# Strings

```
1 def contar_letras(palabra, letra):
2     n = 0
3     r = 0
4     while n < len(palabra):
5         if palabra[n] == letra:
6             r += 1
7         n += 1
8     return r
9
10 a = raw_input("Ingresa algo: ")
11 print a, "tiene", contar_letras(a, "b"), "letras b"
```

```
Ingresa algo: barbudo
barbudo tiene 2 letras b

***Repl Closed***
```

```
print f1 ( '11-24-')
```



```
def f1(x):
    x = int(x)
    if x % 2 == 0:
        return True
    return False

def f2(b):
    i = 0
    while i < 10:
        if str(i) == b:
            return True
        i += 1
    return False

x = '1a2'
i = 0
p = 0
while i < len(x):
    if f2(x[i]):
        if f1(x[i]):
            p += 1
        i += 1

if p > len(x) / 2:
    print 'Sirve'
else:
    print 'No Sirve'
```

[illegible]

The background is a dark blue field filled with various electronic and computing icons. At the top center are two server racks. To the left is a DSLR camera. Below the servers is a desktop monitor displaying a webpage, with a smartphone in front of it. To the right of the monitor is a laptop. Further right is a wireless router. Below the router are crossed wrench and screwdriver tools. In the bottom right is a computer fan. At the bottom center is a keyboard and a mouse. To the left of the keyboard is a hard drive. In the bottom left is a red MP3 player. Other icons include a CD, a game controller, a USB drive, a floppy disk, a webcam, and a printer. White lines connect some of these devices, suggesting a network or data flow.

# Fin de la Ayudantía 3

Gonzalo Fernández

IWI-131 01-10-2018