

# Control de flujo: ciclos (while y for)

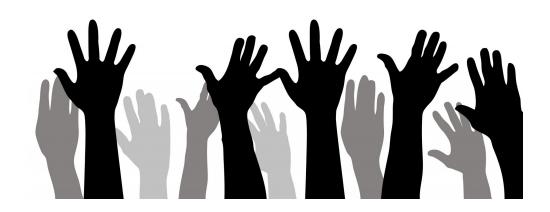
Clase #8

IIC1103 – Introducción a la Programación

## El plan de hoy es...

- Compilado en repl.it
- While! For! ☺
  - Programas complejos, útiles, etc!

### Menti de repaso



#### Problema #1

- Encontrar la mejor nota.
- Si quieres terminar anticipadamente escribe -1
- nota? 5.0
- nota? 4.0
- •
- nota? -1
- mejor= 5.0

#### Problema #2

• Escribe un programa que calcule el dígito verificador del RUT



#### Algoritmo

- 1. Multiplicar cada dígito del RUT por 2, 3, ..., 7, 2, 3, ... de atrás hacia adelante.
- 2. Sumar las multiplicaciones parciales.
- 3. Calcular el resto de la división por 11
- 4. El Dígito Verificador es 11 menos el resultado anterior.
  - Si es 10, se cambia por 'k'.
  - Si es 11, se cambia por 0.

fuente: página de M. Ortega!

#### Algoritmo: Ejemplo

```
• RUT: 11.222.333
```

- 1. 1 1 2 2 2 3 3 3 <-- RUT
- \*3 2 7 6 5 4 3 2 <-- 2, 3, 4, 5, 6, 7, 2, 3, ...</li>
- ------
- 3 2 14 12 10 12 9 6
- 2. SUMA: 3 + 2 + 14 + 12 + 10 + 12 + 9 + 6 = 68
- 3. 68:11=6
- - 66
- ----
- 2 <-- RESTO</li>
- 4. 11 2 = 9 <-- DÍGITO VERIFICADOR

- 1. Multiplicar cada dígito del RUT por 2, 3, ..., 7,
- 2, 3, ... de atrás hacia adelante.
- 2. Sumar las multiplicaciones parciales.
- 3. Calcular el resto de la división por 11
- 4. El Dígito Verificador es 11 menos el resultado anterior.

Si es 10, se cambia por 'k'.

Si es 11, se cambia por 0.

#### Solución – parte 1

```
• rut = int(input("Ingrese su RUT: "))
• m = 2 #multiplicador
\bullet suma = 0
while rut>0:
     ultimo = rut%10 #obtener ultimo dig de RUT
     suma += ultimo*m
     m+=1
     if m==8:
         m=2
     rut = rut//10 #sacar ultimo digito
```

#### Solución – parte 2

```
dv = 11-suma%11
if dv==10:
print("Digito verificador: K")
elif dv==11:
print("Digito verificador: 0")
else:
print("Digito verificador: "+str(dv))
```

#### Problema #3

#### Pregunta 1

Cansado de trabajar, tu compañero te ofrece jugar **Pig**. El juego consiturnos, intentando llegar a los 100 puntos. Si al terminar un turno, el juegador vez que un jugador recibe el dado puede decidir si lo lanzará o no. E

- Si el resultado **no** es 1, entonces el jugador acumula los puntos del reseguir acumulando puntos, o terminar su turno.
- Si el resultado es 1, entonces pierde todos los puntos que había acum

Como no tienen un dado, le ofreces programar el juego de forma que se ver

Puedes usar un diálogo similar al anterior, siempre y cuando se le pregunte al muestre el dado que sale en cada lanzamiento, se muestre los puntajes de ambo finalizar el juego se muestre quién fue el ganador.

```
*** Turno jugador 1
;Lanzar? (1) sí (2) no
>1
Tienes un 6
¿Lanzar? (1) sí (2) no
>1
Tienes un 4
¿Lanzar? (1) sí (2) no
>2
Jugador 1: 10 Jugador 2: 0
*** Turno jugador 2
¿Lanzar? (1) sí (2) no
>1
Tienes un 1
Jugador 1: 10 Jugador 2: 0
```

# Resumen de hoy

#### Hoy vimos

while condicion:
instrucciones

for i in range(n):
 instrucciones

while condicion1:
 instrucciones
 if condicion2:
 break

Comparación de strings:

== : iguales
!= : distintos