DEPARTAMENTO DE
INGENIERÍA
INFORMÁTICA
UNIVERSIDAD DE SANTIAGO DE CHILE

CONTROL DE FLUJO: ESTRUCTURAS DE DECISIÓN

10145 - FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA

10110 – FUNDAMENTOS DE COMPUTACIÓN Y PROGRAMACIÓN

RESUMEN DE CONTENIDOS

OPERADORES DE COMPARACIÓN

- Además de los operadores aritméticos, en Python existen los siguientes **operadores de comparación**

Mayor	`>`	`4.3 > 3.2`
Mayor o igual	`>=`	`4.0 >= 4.1`
Menor	`<`	`-2 < 0`
Menor o igual	`<=`	`-3.14 <= -3.2`
Igual	`==`	`2.0 == 2`
Distinto	`!=`	`-2 != -2.1`

OPERADORES LÓGICOS

- Existen además **operadores lógicos**:
  - La **negación** (`not`) para **invertir** el valor de verdad

```
not 50 > 4 → not True → False
```

  - La **conjunción** o "*y lógico*" (`and`), que resulta verdadero si y solo si **todas** las sub-expresiones son verdaderas

```
x > y and y <= z
```

  - La **disyunción** u "*o lógico*" (`or`), que resulta verdadero si **al menos una** de las sub-expresiones lo es

```
x != y or x <= z or y < z
```

SENTENCIA `if`

- Permite **condicionar la ejecución** de un bloque de sentencias al cumplimiento de una **condición**
  - La condición debe resultar siempre en un booleano
  - Sintaxis:

```
if <condición>:
        # Se ejecuta si la condición se cumple
        <Bloque de sentencias condicionales>
<Bloque de sentencias que sigue>
```

- El bloque condicionado debe estar **indentado**. Para volver al flujo no condicionado es necesario volver al nivel previo de **indentación**

SENTENCIA `if-else`

- Permite **dividir el flujo de la ejecución de dos bloques** de sentencias según el cumplimiento de una **condición**
  - Sintaxis:

```
if <condición>:
    # Se ejecuta si la condición se cumple
    <Bloque de sentencias condicionales>
else:
    # Se ejecuta si la condición no se cumple
    <Bloque de sentencias alternativo>
# Se ejecuta independiente de si la condición se cumplió o no
<Bloque de sentencias que sigue>
```

SENTENCIA if-elif-else

- Sintaxis:

```
if <condición 1>:
    # Se ejecuta si la condición 1 se cumple
    <Bloque de sentencias condicionales 1>
elif <condición 2>:
    # Se ejecuta si la condición 1 no se cumple y sí se cumple la
condición 2 (Se puede crear tantos bloques elif como sea necesario,
cada uno con una condición distinta)
    <Bloque de sentencias condicionales 2>
else:
    # Se ejecuta si las condiciones 1 y 2 no se cumplen
    <Bloque de sentencias alternativo>
# Se ejecuta independiente de si la condición se cumplió o no
<Bloque de sentencias que sigue>
```

EJERCICIOS

EJERCICIO PROPUESTO

- Construye un programa que solicite el valor de dos ángulos interiores de un triángulo e indique si un triángulo es equilátero, isósceles o escaleno

  - Recuerda que los ángulos interiores siempre sumarán 360
  - Las entradas siempre serán números enteros positivos con valores de ángulos válidos.

EJERCICIO PROPUESTO

- Construye un programa que reciba una cantidad de dinero y entregue el desglose de este en billetes y monedas válidos en circulación en Chile (20.000, 10.000, 5.000, 2.000, 1.000, 500, 100, 50, 10, 5, 1)

- Por ejemplo, si la entrada fuese $20.250, el programa debería entregar:
  - 1 billete(s) de $20.000
  - 2 moneda(s) de $100
  - 1 moneda(s) de $50

TAREAS PARA TRABAJO AUTÓNOMO

1. Revisar el apunte:
   - Tipos de datos, operadores y expresiones en Google Colab (Disponible en: https://github.com/PROGRA-FING-USACH/2023-1/blob/main/Lecturas/04_iteracion_-_while.ipynb)

2. Revisar la guía 2, disponible a partir del lunes en Replit

¿CONSULTAS?