



El futuro digital
es de todos

MinTIC

Ciclo 1

Fundamentos de programación con Python

Sesión 2: Principios de pensamiento computacional

Programa Ciencias de la Computación e Inteligencia Artificial
Escuela de Ciencias Exactas e Ingeniería
Universidad Sergio Arboleda
Bogotá



UNIVERSIDAD
SERGIO ARBOLEDA





Agenda

1. Principios de pensamiento computacional
2. Abstracción
3. Generalización
4. Modelos
5. Reconocimiento de patrones
6. Pensamiento algorítmico
7. Estructuras de control





1. Principios Pensamiento Computacional

- Abstracción y representación.
- Modelamiento: Denotación.
- Descomposición.
- Generalización y reconocimiento de Patrones.
- Pensamiento lógico y algorítmico.
- Pensamiento deductivo
- Pensamiento inductivo





2. Abstracción y Representación



(a) Transbordador



(b) Quetzalcóatl

Las habilidades cognitivas han permitido al hombre:

- a) El lanzamiento de transbordadores espaciales que colocan satélites en la órbita de la tierra.
- b) La creación de deidades religiosas que simbolizan creencias y prácticas del tipo existencial y moral.





2. Abstracción y Representación

- La abstracción es el filtro utilizado para quedarse con lo que se considera esencial, eliminando toda la complejidad innecesaria.
- La abstracción es la habilidad que le permite al ser humano combatir la complejidad al considerar sólo lo esencial del objeto o fenómeno que se esté analizando.





2. Abstracción y Representación

El conocimiento humano está dividido en diferentes áreas de estudio porque cada una se enfoca en un aspecto específico de la realidad, cada disciplina tiene sus propias abstracciones. En ese sentido, la abstracción es un concepto clave en toda actividad humana y en cualquier área de estudio como: la matemática, la física, la biología o el arte.





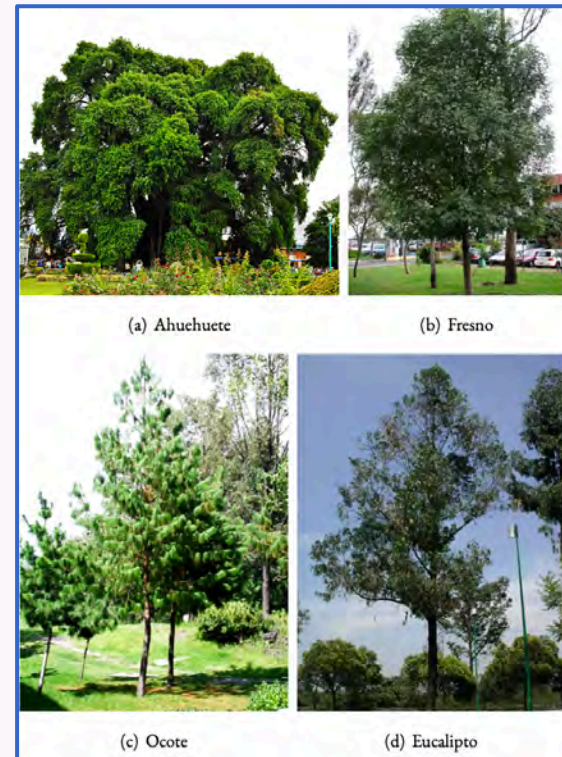
3. Generalización

La generalización es el proceso de formular conceptos genéricos a través de la extracción de cualidades comunes de ejemplos concretos.

Ejemplo 1: Árboles

Ejemplo 2: Ley de la gravitación universal de Newton

$$F = G \frac{m_1 m_2}{d^2}$$





3. Generalización

- **Eliminación de detalles:** es el proceso de dejar fuera de consideración una o más propiedades de un objeto con la finalidad de enfocarse sólo en algunas propiedades.
- **Niveles de abstracción:** se refiere al grado de detalle con el que se especifica una representación. Una representación en un alto nivel de abstracción especifica menos detalles que una representación en un bajo nivel de abstracción.





4. Modelos

Partiendo de que la abstracción es una habilidad esencial para la construcción de modelos y la descomposición de problemas, se define modelo como:

Una representación abstracta (matemática, declarativa, visual, etc.) de fenómenos, sistemas o procesos. El ser humano crea modelos que representan la esencia de determinados fenómenos a fin de analizarlos y comprenderlos.





4. Modelos

- **Modelo declarativo:** todo cuerpo atrae a los demás con una fuerza que, para dos cuerpos cualesquiera, es directamente proporcional al producto de sus masas e inversamente proporcional al cuadrado de la distancia que los separa.
- **Modelo matemático:** $F = G \frac{m_1 m_2}{d^2}$





5. Reconocimiento de Patrones

- El área de reconocimiento de patrones concentra algoritmos, metodologías, teorías y sistemas que permiten diferenciar, asociar y clasificar datos de forma automatizada.
- En la vida diaria ya convivimos con tecnología que reconoce nuestra voz, nuestra cara o nuestra huella digital. Además, también se aplica el reconocimiento de patrones cuando buscamos en la red imágenes e información.





Ejercicio 1.

- Imagina que eres un desarrollador de software y tienes que crear un personaje de un videojuego de pelea.
- ¿Qué propiedades y comportamientos de tu luchador serían relevantes?
- Diseña un cubo mágico diferente de 3×3 . Puedes construir uno de 4×4 .

4	9	2	15
3	5	7	15
8	1	6	15
15	15	15	15





Ejercicio 2.

- Tenemos que pronosticar qué tan lejos puede viajar una pelota si se le aplica una fuerza determinada. Si la prueba es en el aire y en el agua:
 - ¿Qué elementos tenemos que considerar y porqué?
 - ¿Qué pasa si la prueba es en el espacio?





6. Pensamiento Algorítmico

Un algoritmo es una serie de pasos ordenados que se siguen para resolver un problema. En la vida cotidiana se emplean algoritmos ampliamente, por ejemplo, en las recetas para preparar platillos.

Ejemplo: Un algoritmo para preparar un pastel de queso.





6. Pensamiento Algorítmico



1. Mezclar en una licuadora 100gr de galletas María y 50gr de mantequilla hasta que quede una pasta de galletas parecida a la arena mojada.
2. Aplastar la pasta de galletas en un recipiente hondo hasta que quede una base compacta.
3. Colocar el recipiente hondo en un refrigerador por 15 minutos.
4. Añadir en una cazuela 500gr de queso crema, 500gr de nata, 100gr de azúcar y 15gr de gelatina en polvo.
5. Mezclar los ingredientes anteriores a fuego lento mientras haya grumos.
6. Añadir la mezcla sobre la base de galletas del recipiente hondo.
7. Colocar el recipiente hondo en el refrigerador por 4 horas.
8. Cubrir la superficie de la tarta con 300gr de mermelada de fresa.





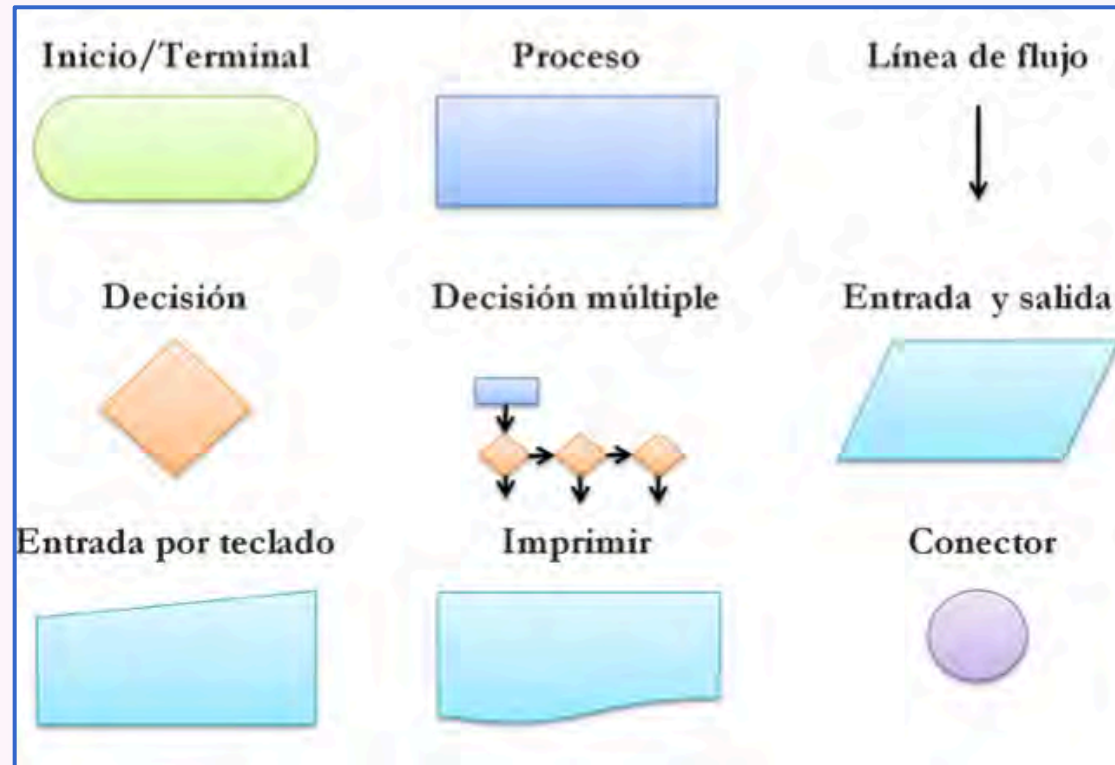
6. Pensamiento Algorítmico

Al igual que en las recetas, un problema en particular, frecuentemente puede resolverse de varias maneras. Por esta razón, hay varios algoritmos que pueden solucionarlo con diferentes ventajas y desventajas, algunos tardan más tiempo en encontrar una solución u otros consumen más memoria.



6. Pensamiento Algorítmico

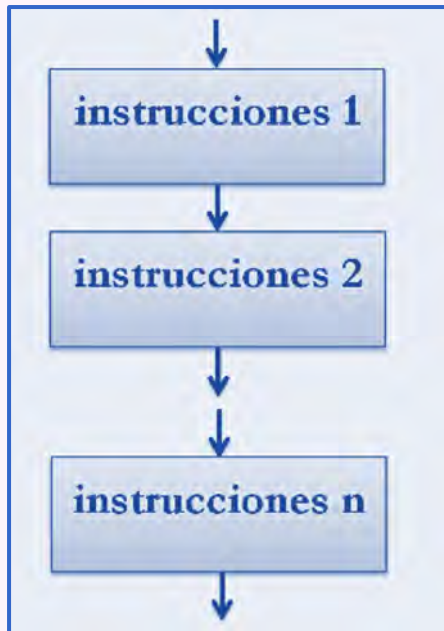
- Diagramas de flujo
- Seudocódigo
- Variables
- Contadores
- Acumuladores



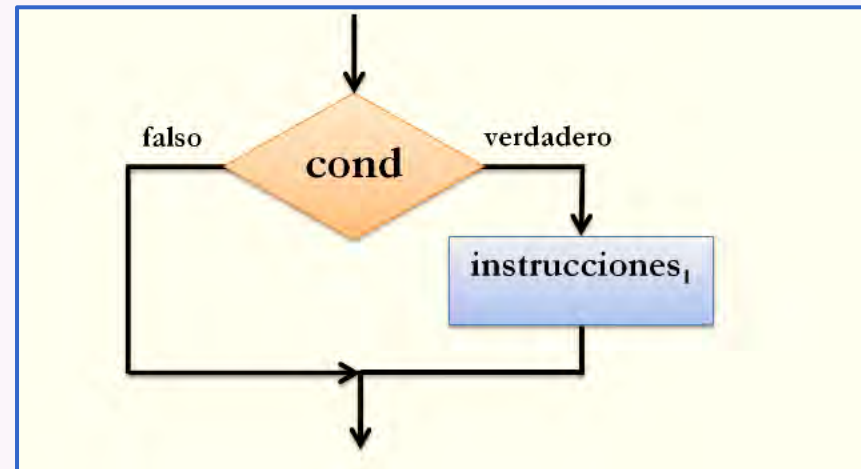


7. Estructuras de control

Estructura de secuencia



Estructura de decisión

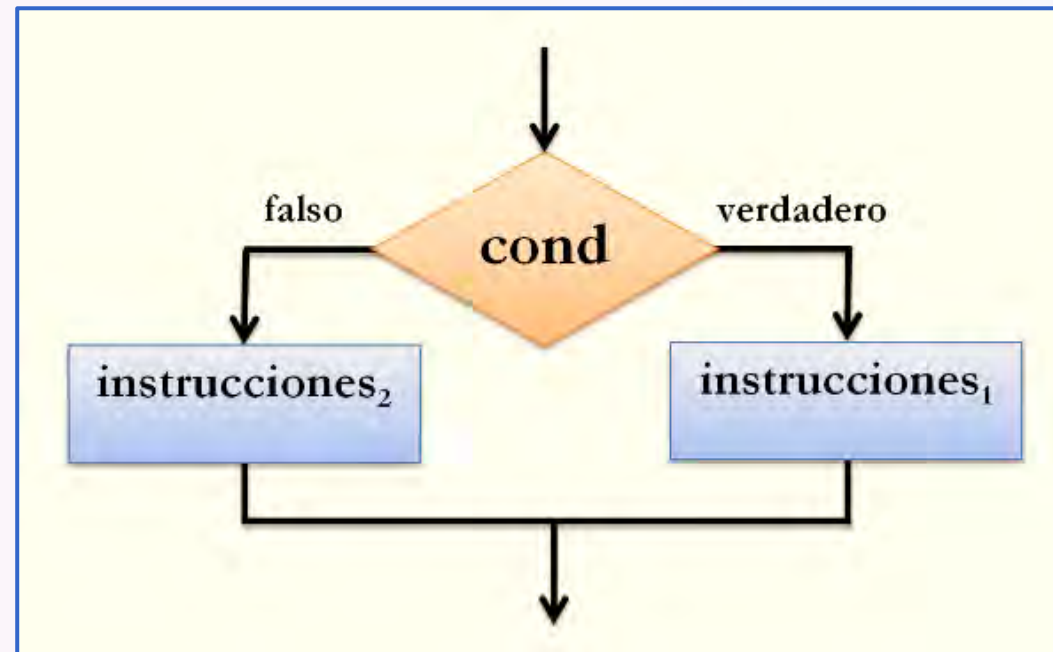




7. Estructuras de control

- Decisión Disyuntiva

```
1: si condición entonces  
2:   instrucciones1  
3: si no  
4:   instrucciones2  
5: fin si
```

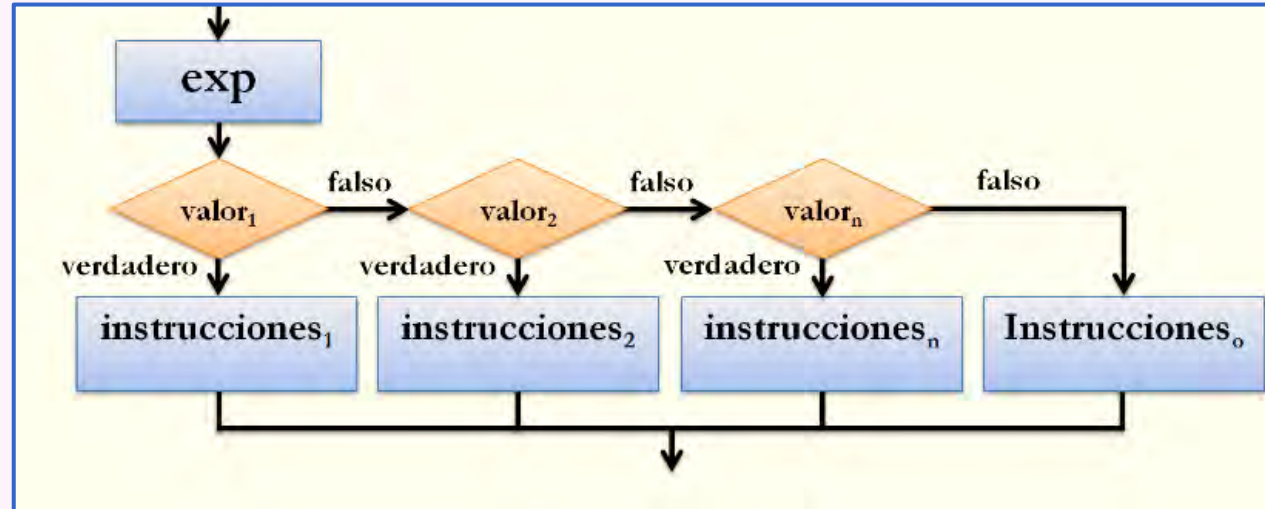




7. Estructuras de control

- Decisión múltiple

```
1: seleccionar expresión  
2: caso valor1  
3:   instrucciones1  
4: caso valor2  
5:   instrucciones2  
6: caso valorn  
7:   instruccionesn  
8: caso por omisión  
9:   instruccioneso  
10: fin seleccionar
```



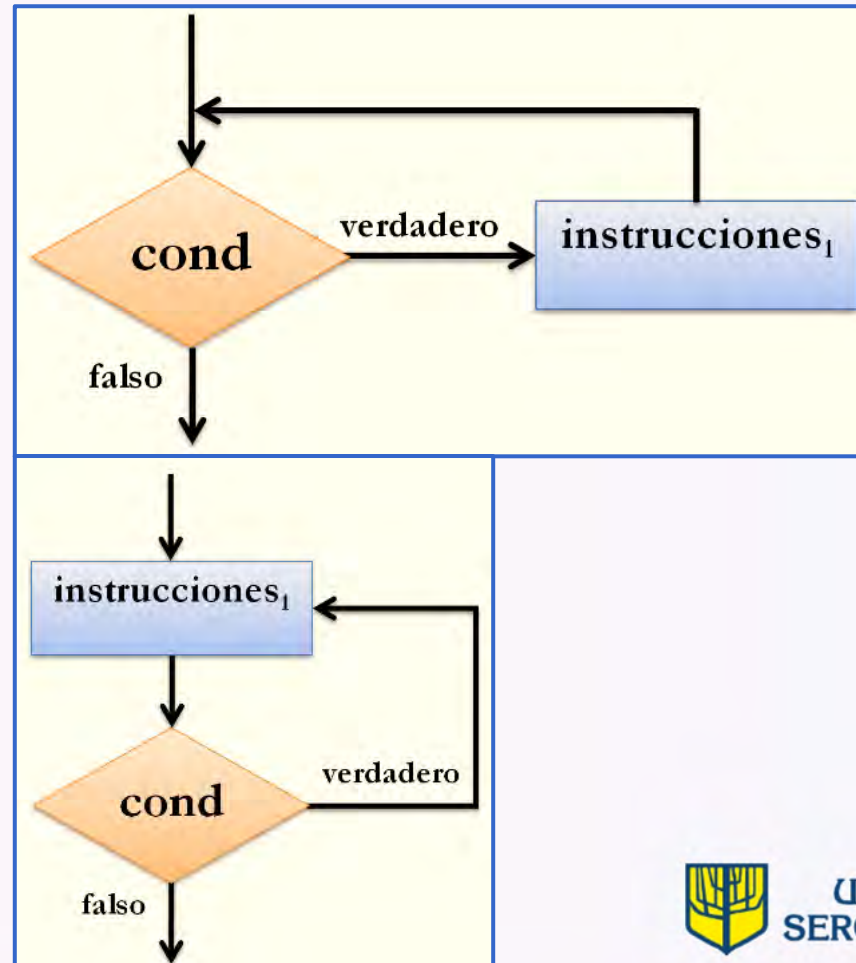


7. Estructuras de control

- Estructura de repetición

1: **mientras** *condición* **hacer**
2: *instrucciones₁*
3: **finmientras**

1: **repetir**
2: *instrucciones₁*
3: **hasta que** *condición*

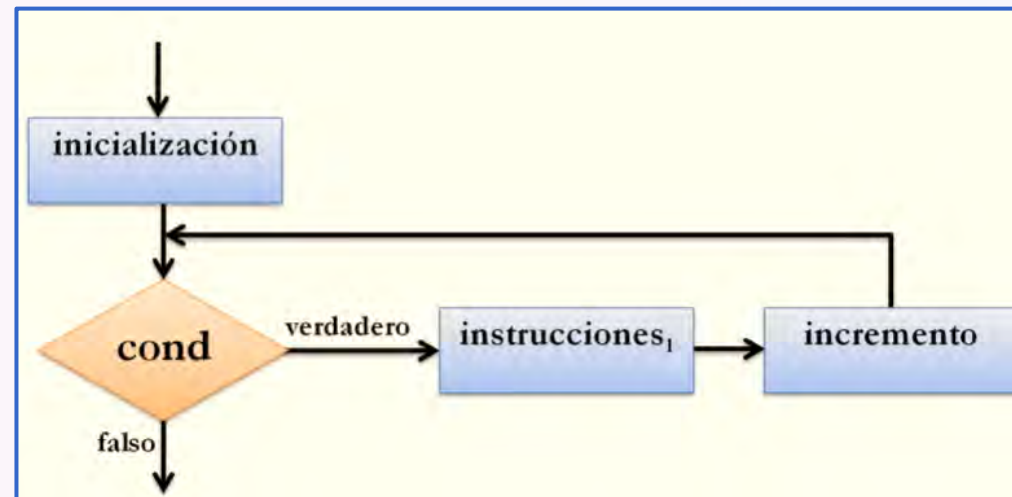




7. Estructuras de control

- Estructura repetición para

```
1: para inicialización; condición; incremento hacer  
2:   instrucciones1  
3: finpara
```





Ejercicio 3.

- Un profesor desea felicitar a sus mejores estudiantes y motivar a los otros a enfocarse más en las materias. Los mejores estudiantes son aquellos que tienen un promedio igual o mayor que 9. El grupo tiene 20 estudiantes.
- Diseña un algoritmo en pseudocódigo para ayudarle al profesor a cumplir con su objetivo. Utiliza la estructura de control "repetir para".





Ejercicio 4.

- El auto de un amigo de Tulio tiene una llanta pinchada y quieren cambiarla, pero no saben cómo.
- Diseña un algoritmo en pseudocódigo para ayudarlos a cambiar la llanta pinchada.





Preguntas

