

INTRODUCCIÓN A LA LÓGICA Y ALGORITMOS

Hernández Pardo Valerie Jireth
Presenta a: Karen de los rios

Universidad Distrital Francisco Jose de Caldas
Logaritmos y programación en C++
2025

Estructura del argumento y lógica formal e informal

■ Actividad 1: Identificación de argumentos

1. Si llueve, entonces la calle está mojada. Ha llovido, por lo tanto, la calle está mojada.
2. Todos los perros ladran. Rex es un perro. Por lo tanto, Rex ladra.
3. María estudia mucho, por lo que debe ser una persona inteligente.

PREMISAS	CONCLUSIÓN	VALIDEZ
-Si llueve, entonces la calle está mojada. -Ha llovido.	-La calle esta mojada.	Es válido, ya que es un argumento deductivo. Entonces si las premisas son verdaderas, la conclusión debe ser verdadera.
-Todos los perros ladran. -Rex es un perro.	-Rex ladra.	Es válido, debido a que de una premisa universal y un caso particular se deduce la conclusión.
-María estudia mucho.	-María debe ser una persona inteligente.	El argumento no es válido porque la conclusión no se sigue de manera necesaria de la premisa.

■ Actividad 2: Evaluación de argumentos

1. Todos los gatos son mamíferos. Todos los mamíferos vuelan. Por lo tanto, todos los gatos vuelan.

- **Premisa 1:** Todos los gatos son mamíferos: Verdadera.
- **Premisa 2:** Todos los mamíferos vuelan: Falsa.
- **Conclusión:** Todos los gatos vuelan: Falsa.

La estructura lógica es válida (si las premisas fueran verdaderas, la conclusión lo sería), en este caso vemos una premisa falsa, por lo tanto todo el argumento es incorrecto.

2. Si Juan estudia, aprobará el examen. Juan estudió. Por lo tanto, aprobó el examen.

- **Premisa 1:** Si Juan estudia, aprobará el examen: Verdadera.
- **Premisa 2:** Juan estudió: Verdadera.
- **Conclusión:** Juan aprobó el examen: Verdadera.

La estructura lógica es válida, como las premisas son verdaderas entonces la conclusión de que aprobó el examen es verdadera.

Proposiciones lógicas y operadores lógicos

■ Actividad 3: Tablas de verdad

- ¿Qué es una tabla de verdad?
- Una tabla de verdad muestra todas las posibilidades de verdad/falsedad de un argumento y nos ayuda a comprobar su validez.
- P: "Hoy es lunes" y Q: "Está soleado".
- V (verdadero), F (falso)

1. $P \wedge Q$

P	Q	$P \wedge Q$
V	V	V
V	F	F
F	V	F
F	F	F

2. $P \vee Q$

P	Q	$P \vee Q$
V	V	V
V	F	V
F	V	V
F	F	F

3. $\neg P \rightarrow Q$

P	Q	$\neg P$	$\neg P \rightarrow Q$
V	V	F	V
V	F	F	V
F	V	V	V
F	F	V	F

4. $P \leftrightarrow Q$

P	Q	$P \leftrightarrow Q$
V	V	V
V	F	F
F	V	F
F	F	V

Conceptos básicos y pseudocódigo

■ Actividad 4: ¿Qué es un algoritmo?

- Para mí, un algoritmo es un conjunto de instrucciones claras y ordenadas que permiten resolver un problema o realizar una tarea paso a paso, con el fin de llegar a un resultado.
- En la vida cotidiana podemos encontrar algunos algoritmos que aunque no nos demos cuenta ahí están por ejemplo:
 1. Una receta de cocina
 2. Sembrar una planta
 3. Ir a la universidad

La programación estructurada es importante porque permite escribir programas claros y ordenados, organizados en pasos lógicos que facilitan su comprensión. Además, ayuda a detectar y corregir errores con mayor facilidad, simplifica el mantenimiento y modificación del código, y fomenta la reutilización de partes ya creadas. En conjunto, hace que el desarrollo de software sea más eficiente y confiable.

■ Actividad 5: Etapas del desarrollo de un programa

- **Análisis del problema:** Consiste en comprender qué se quiere resolver, qué datos se tienen y qué resultado se espera.
- **Diseño del algoritmo:** Escribir los pasos que debe seguir el programa para resolver el problema, de manera clara y ordenada (como una receta).
- **Codificación:** Traducir el algoritmo a un lenguaje de programación específico (Python, Java, C++, etc.).
- **Compilación y ejecución:** Transformar el código en instrucciones que la computadora pueda entender y ponerlo en marcha para ver si funciona.
- **Verificación y depuración:** Revisar que el programa haga lo que debe hacer, corregir errores (bugs) y mejorar el funcionamiento.
- **Documentación:** Redactar explicaciones sobre el funcionamiento del programa, cómo usarlo y cómo fue construido, para que otros (o el mismo programador) puedan entenderlo en el futuro.

Representación de algoritmos y estructuras de control

■ Actividad 6: Pseudocódigo y diagramas de flujo

• Pseudocódigo

```
# Programa que compara dos numeros
numero1 = int(input("Ingresa el primer numero: "))
numero2 = int(input("Ingresa el segundo numero: "))
if numero1 > numero2:
    print("El primer numero es mayor:", numero1)
elif numero2 > numero1:
```

```

    print("El segundo numero es mayor:", numero2)
else:
    print("Ambos numeros son iguales")

```

■ Diagrama de flujo

- Inicio
- Ingresar números 1 y 2
- ¿número1¿número2?
- Sí
- Imprimir .^{El} primer número es mayor”
- No
- ¿número2¿número1?
- Sí
- Imprimir .^{El} segundo número es mayor”
- No
- Imprimir .^{Ambos} números son iguales”
- Fin

■ Actividad 7: Uso de estructuras de control

```

# Programa que determina si un numero es par o impar
# Paso 1: Solicitar un numero
numero = int(input("Ingrese un numero: "))
# Paso 2 y 3: Verificar si es par o impar
if numero % 2 == 0:
    print("El numero es par")
else:
    print("El numero es impar")

```

■ Actividad 8: Estructuras de repetición

- Pseudocódigo que sume los números del 1 al 20

```

suma = 0
for i in range(1, 21):
    suma += i
print("La suma de los numeros del 1 al 20 es:", suma)

```

- Y para que sume los pares del 1 al 20 se usa el código:

```

suma = 0
for i in range(1, 21):
    if i % 2 == 0:
        suma += i
print("La suma de los numeros pares del 1 al 20 es:", suma)

```

Cierre y Reflexión

1. ¿Qué fue lo más fácil y lo más difícil del taller?
 - Lo más fácil fue identificar ejemplos cotidianos de algoritmos, porque se parecen a actividades comunes como recetas o rutinas.
 - Lo más difícil fue entender cómo se aplican las tablas de verdad y la relación con la validez de los argumentos, ya que requieren más análisis lógico.
2. ¿Cómo se relaciona la lógica con la programación? La lógica es la base de la programación, porque permite tomar decisiones, analizar condiciones y seguir pasos ordenados. Gracias a la lógica, un programa puede ejecutar instrucciones correctas, resolver problemas y dar resultados coherentes.
3. ¿Qué aplicaciones prácticas pueden tener los algoritmos en su vida diaria? Los algoritmos están presentes en muchas actividades cotidianas: desde preparar una receta, seguir una ruta en Google Maps, organizar la rutina diaria, hasta usar aplicaciones como redes sociales o buscadores. En general, cualquier tarea que se haga paso a paso puede verse como un algoritmo.

Referencias bibliográficas

- Componentes del razonamiento: premisas y conclusión. (2023). Objetos de aprendizaje UNAM. Recuperado de <http://objetos.unam.mx/logica/premisasConclusion/index.html>
- Las tablas de la verdad. (s.f.). REA Las aventuras de Filoland. Proyecto EDIA. Filosofía. Bachillerato. Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado (INTEF), Ministerio de Educación. https://descargas.intef.es/cedec/proyectoedia/contenidos/Filoland3_palabras_poder/las_tablas_de_la_verdad.pdf
- ¿Qué es el Pseudocódigo y Cómo Puede Mejorar tu Programación? (2025, 26 de febrero). Kinsta, Centro de Recursos. Recuperado de <https://kinsta.com/es/base-de-conocimiento/que-es-pseudocodigo/>
- Explicación de algoritmo y diagrama de flujo con ejemplos. (2025, 19 de febrero). Edraw. Recuperado de <https://www.edrawsoft.com/es/explain-algorithm-flowchart.html>