

Introducción a la Lógica y Algoritmos

Yuliza Estupiñan

Código: 20241135056

Universidad Distrital Francisco José de Caldas

Septiembre 2025

PARTE 1

ACTIVIDAD 1

Enunciado: “Si llueve, entonces la calle está mojada. Ha llovido, por lo tanto, la calle está mojada.”

Premisas:

- Si llueve, la calle está mojada.
- Ha llovido.

Conclusión: La calle está mojada.

Evaluación: El argumento es válido, porque si ambas premisas son verdaderas, necesariamente la conclusión también lo es. Este es un caso clásico de razonamiento correcto.

1.1

Enunciado: “Todos los perros ladran. Rex es un perro. Por lo tanto, Rex ladra.”

Premisas:

- Todos los perros ladran.
- Rex es un perro.

Conclusión: Rex ladra.

Evaluación: El argumento es válido, porque de la idea de que todos los perros ladran y que Rex es un perro, se sigue necesariamente que Rex ladra. Su solidez depende de la verdad de la primera premisa: en la realidad, no todos los perros ladran.

1.2

Enunciado: “María estudia mucho, por lo que debe ser una persona inteligente.”

Premisa: María estudia mucho.

Conclusión: María es inteligente.

Evaluación: El argumento no es deductivamente válido, porque que alguien estudie mucho no garantiza que sea inteligente. Puede ser cierto en algunos casos, pero no necesariamente en todos. Se trata más bien de un argumento de tipo inductivo: la premisa hace probable la conclusión, pero no la asegura.

Conclusión general

Los dos primeros casos son ejemplos de argumentos deductivos válidos; su solidez depende de la verdad de las premisas. El tercer caso es un argumento informal de tipo inductivo que no garantiza la conclusión.

ACTIVIDAD 2: Evaluación de argumentos

1.1

Enunciado: “Todos los gatos son mamíferos. Todos los mamíferos vuelan. Por lo tanto, todos los gatos vuelan.”

Premisas:

- Todos los gatos son mamíferos. — (Verdadera)
- Todos los mamíferos vuelan. — (Falsa)

Conclusión: Todos los gatos vuelan. — (Falsa)

1. Identifico las premisas explícitas: la primera dice que la clase de los gatos está incluida en la clase de los mamíferos; la segunda dice que la clase de los mamíferos está incluida en la clase de los que vuelan.
2. Formalmente (por la forma lógica), si ambas premisas fuesen verdaderas, por transitividad la conclusión se seguiría: los gatos estarían incluidos entre los que vuelan, por tanto todos los gatos volarían. Es decir, la forma del argumento es deductiva y válida.
3. En la realidad, la primera premisa es verdadera (los gatos sí son mamíferos) y la segunda es falsa (la mayoría de mamíferos no vuelan). Por eso la conclusión es falsa.
4. Resultado: el argumento es válido en su forma lógica, pero no es sólido porque contiene una premisa falsa. (Validez = forma correcta; solidez = validez + premisas verdaderas.)

1.2

Enunciado: “Si Juan estudia, aprobará el examen. Juan estudió. Por lo tanto, aprobó el examen.”

Premisas:

- Si Juan estudia, aprobará el examen. — (Condicional: su verdad depende del contexto y de cuánto significa “estudiar” o la dificultad del examen.)
- Juan estudió. — (Hecho dado.)

Conclusión: Juan aprobó el examen.

1. Formalmente, si la premisa condicional es verdadera y la afirmación “Juan estudió” es verdadera, entonces la conclusión necesariamente es verdadera. Así que la forma es válida.
2. En la práctica, la veracidad de la conclusión depende de si la premisa condicional refleja la realidad (¿estudiar garantiza aprobar en ese contexto?). Si la condicional es sólida (verdadera), entonces el argumento es sólido; si no, el argumento aunque válido no es sólido.
3. Resultado: el argumento es válido (forma correcta) y su solidez depende de la verdad de la premisa condicional.

ACTIVIDAD 3: Propositiones lógicas y operadores — Tablas de verdad

1.1

¿Qué es una tabla de verdad?

Una tabla de verdad muestra todas las combinaciones posibles de valores de verdad (Verdadero / Falso)

de las proposiciones simples y calcula el valor de verdad de una expresión compuesta para cada combinación. Sirve para analizar cómo se comportan los conectivos lógicos (y, o, no, implica, si y solo si).

Considera:

P: “Hoy es lunes”

Q: “Está soleado”

1.2

Tabla para las expresiones pedidas: $(P \wedge Q)$, $(P \vee Q)$, $(\neg P \rightarrow Q)$, $(P \leftrightarrow Q)$.

Tabla de verdad (todas las expresiones juntas):

P	Q	$(P \wedge Q)$	$(P \vee Q)$	$(\neg P \rightarrow Q)$	$(P \leftrightarrow Q)$
V	V	V	V	V	V
V	F	F	V	V	F
F	V	F	V	V	F
F	F	F	F	F	V

Explicación fila por fila

P = V, Q = V

- $P \wedge Q$: verdadero \wedge verdadero \rightarrow V.
- $P \vee Q$: verdadero \vee verdadero \rightarrow V.
- $\neg P \rightarrow Q$: $\neg P = F$, antecedente falso \rightarrow V.
- $P \leftrightarrow Q$: ambos tienen mismo valor (V y V) \rightarrow V.

P = V, Q = F

- $P \wedge Q$: V \wedge F \rightarrow F.
- $P \vee Q$: V \vee F \rightarrow V.
- $\neg P \rightarrow Q$: $\neg P = F$, antecedente falso \rightarrow V.
- $P \leftrightarrow Q$: V \leftrightarrow F \rightarrow F.

P = F, Q = V

- $P \wedge Q$: F \wedge V \rightarrow F.
- $P \vee Q$: F \vee V \rightarrow V.
- $\neg P \rightarrow Q$: $\neg P = V$, y Q = V \rightarrow V.
- $P \leftrightarrow Q$: F \leftrightarrow V \rightarrow F.

P = F, Q = F

- $P \wedge Q$: F \wedge F \rightarrow F.
- $P \vee Q$: F \vee F \rightarrow F.
- $\neg P \rightarrow Q$: $\neg P = V$, pero Q = F \rightarrow F.
- $P \leftrightarrow Q$: F \leftrightarrow F \rightarrow V.

PARTE 2

Actividad 1: ¿Qué es un algoritmo?

Definición

Un algoritmo es un conjunto de pasos o instrucciones organizadas de manera lógica y ordenada que sirven para resolver un problema o realizar una tarea. Se parecen mucho a una receta de cocina porque nos dicen qué hacer primero, después y al final, hasta llegar al resultado esperado.

Ejemplos de algoritmos en la vida cotidiana

1. Seguir una receta para preparar un plato de comida.
2. Los pasos para vestirse en la mañana (primero elegir la ropa, luego ponérsela, después abrocharse los zapatos).
3. El proceso para cruzar una calle con semáforo (esperar luz verde, mirar a ambos lados, caminar).

Importancia de la programación estructurada

La programación estructurada es importante porque nos permite organizar los algoritmos de una forma clara, fácil de entender y sin confusiones. Gracias a esto, se evitan errores, se facilita la lectura del código y se puede modificar o mejorar sin tantos problemas. Además, hace que las soluciones sean más eficientes y reutilizables.

Actividad 2: Etapas del desarrollo de un programa

Análisis del problema

Es la etapa en la que se entiende bien qué se quiere resolver. Aquí se identifican los datos de entrada, lo que se espera obtener como salida y las condiciones que deben cumplirse.

Diseño del algoritmo

Consiste en organizar los pasos que se van a seguir para resolver el problema. Se puede representar con diagramas de flujo o con pseudocódigo, de manera ordenada y lógica.

Codificación

En esta etapa se traduce el algoritmo a un lenguaje de programación para que la computadora lo pueda interpretar.

Compilación y ejecución

Se transforma el código fuente en un programa ejecutable y luego se pone en marcha para comprobar cómo funciona.

Verificación y depuración

Aquí se revisa que el programa funcione correctamente y se corrigen los errores o fallos que aparezcan durante la ejecución.

Documentación

Es la etapa donde se registra y explica cómo está hecho el programa, qué hace y cómo se debe usar, para que sea más fácil de entender por otras personas o para futuras mejoras.

Actividad 2

Pseudocódigo

```
Inicio
  Leer num1
  Leer num2
  Si num1 > num2 Entonces
    Escribir "num1 es mayor"
  Sino
    Si num2 > num1 Entonces
      Escribir "num2 es mayor"
    Sino
      Escribir "Ambos números son iguales"
  FinSi
FinSi
Fin
```

Diagrama de flujo

Actividad 3

Pseudocódigo: Número par o impar

```
Inicio Escribir "Ingrese un número entero:"Leer num
resto ← num MOD 2
Si resto = 0 Entonces Escribir ".El número es par"Sino Escribir ".El número es impar"FinSi Fin
```

Actividad 4

1. Suma de los números del 1 al 20

```
Inicio suma ← 0 Para i ← 1 Hasta 20 Hacer suma ← suma + i FinPara Escribir "La suma de los números del 1 al 20 es: ", suma Fin
```

2. Suma de los números pares del 1 al 20

```
Inicio suma ← 0 Para i ← 1 Hasta 20 Hacer Si i MOD 2 = 0 Entonces suma ← suma + i FinSi FinPara Escribir "La suma de los números pares del 1 al 20 es: ", suma Fin
```

Cierre y Reflexión

- ¿Qué fue lo más fácil y lo más difícil del taller?

Lo más fácil del taller fue entender los ejemplos de la vida cotidiana que se parecen a un algoritmo, porque son cosas que hacemos todos los días sin darnos cuenta, como seguir pasos para cocinar o para llegar a un lugar.

Lo más difícil fue organizar bien el pseudocódigo y los diagramas de flujo, ya que requieren pensar de forma ordenada y lógica, y a veces me confundía con las condiciones o los bucles.

- **¿Cómo se relaciona la lógica con la programación?**

La lógica es la base de la programación, porque nos ayuda a estructurar las ideas y a tomar decisiones correctas dentro de un programa. Sin lógica sería imposible que el programa siguiera un orden, evaluara condiciones o repitiera procesos.

- **¿Qué aplicaciones prácticas pueden tener los algoritmos en su vida diaria?**

Los algoritmos están presentes en muchas cosas de la vida diaria. Por ejemplo, en la forma en que uso el celular (cuando desbloqueo, sigo pasos), en la organización de mis tareas (ordenar actividades con prioridades), o incluso en actividades simples como hacer una receta o planear el transporte para llegar a un lugar. En todo momento estamos aplicando pasos lógicos que son como pequeños algoritmos.