



unab

**UNIVERSIDAD NACIONAL
GUILLERMO BROWN**

Introducción a la Programación

Algoritmos y Estructuras de Datos

-00184-

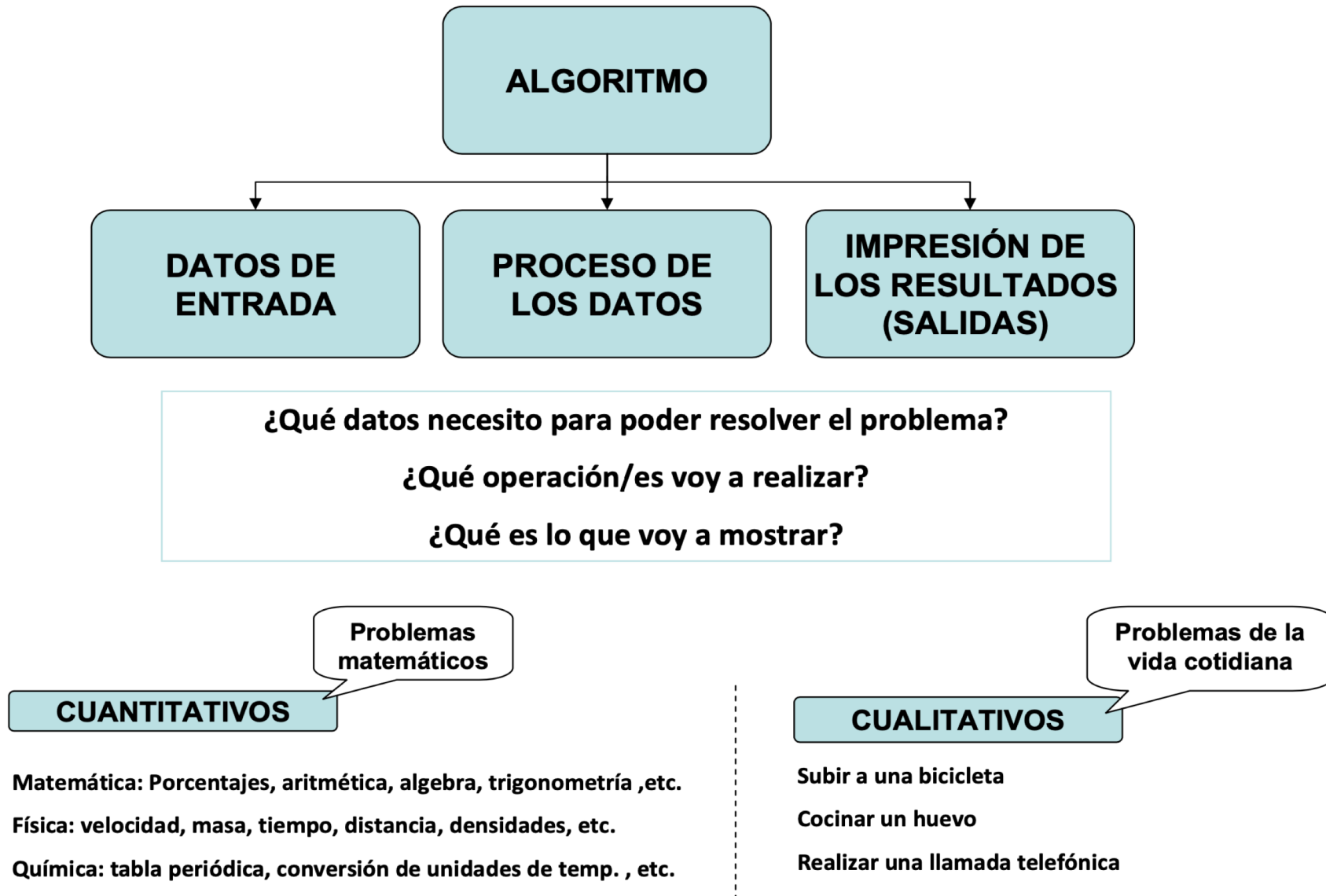
Dr. Diego Agustín Ambrossio

Anl. Sis. Angel Leonardo Bianco

Contenido:

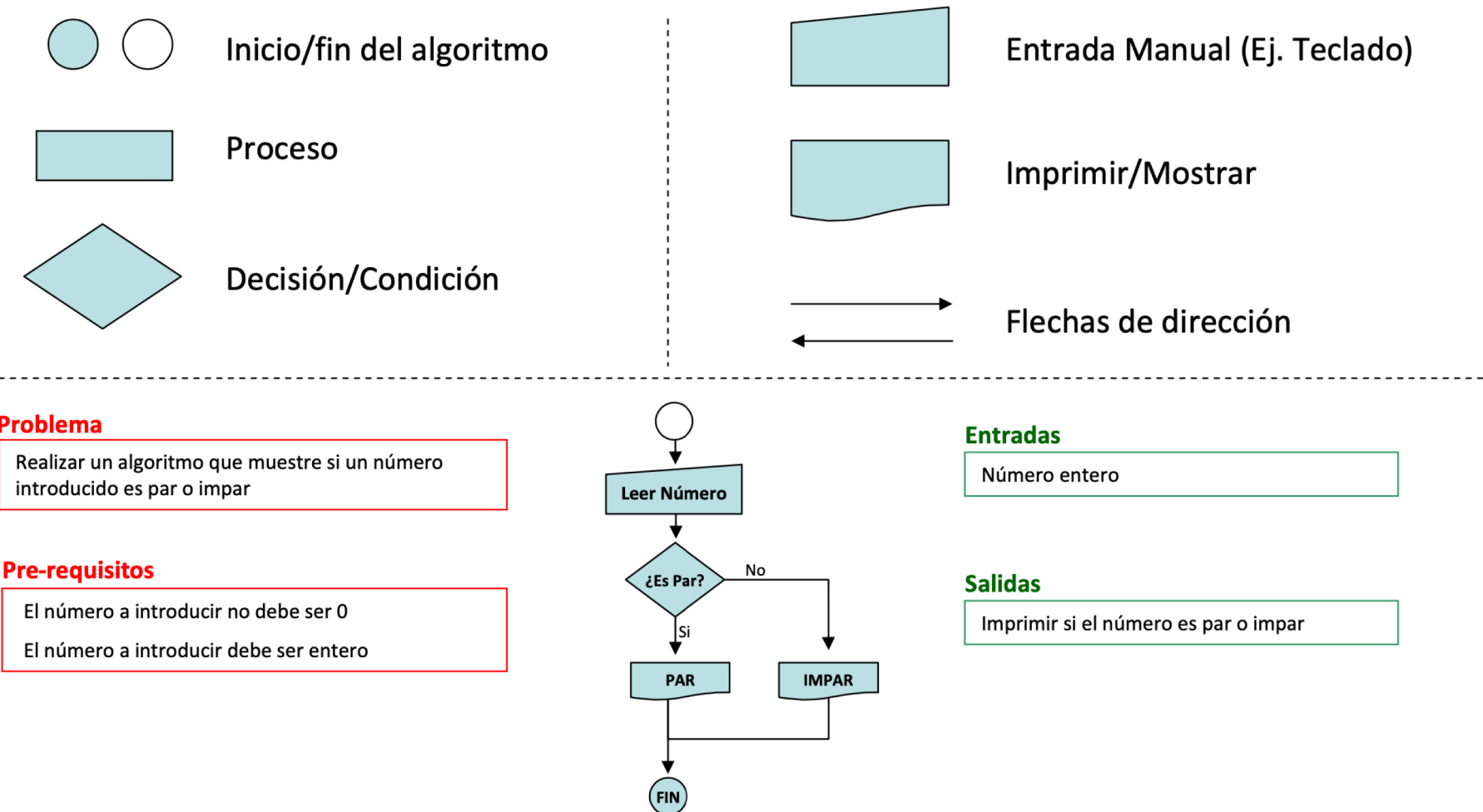
- Algoritmos (Cont.)
- Diagramas de Flujo
- Pseudocódigo
 - Asignación
 - Secuencia
 - Decisión
 - Repetición
 - Selección
 - Iteración

Algoritmos:



Diagramas de Flujo:

Un **diagrama de flujo** representa la esquematización gráfica de un algoritmo. Muestra gráficamente los pasos a seguir para alcanzar resolución de un problema.



Pseudocódigo:

El **pseudocódigo**, es una serie de palabras léxicas y gramaticales referidos a los lenguajes de programación, pero sin llegar a la rigidez de la sintaxis de estos ni a la fluidez del lenguaje coloquial.

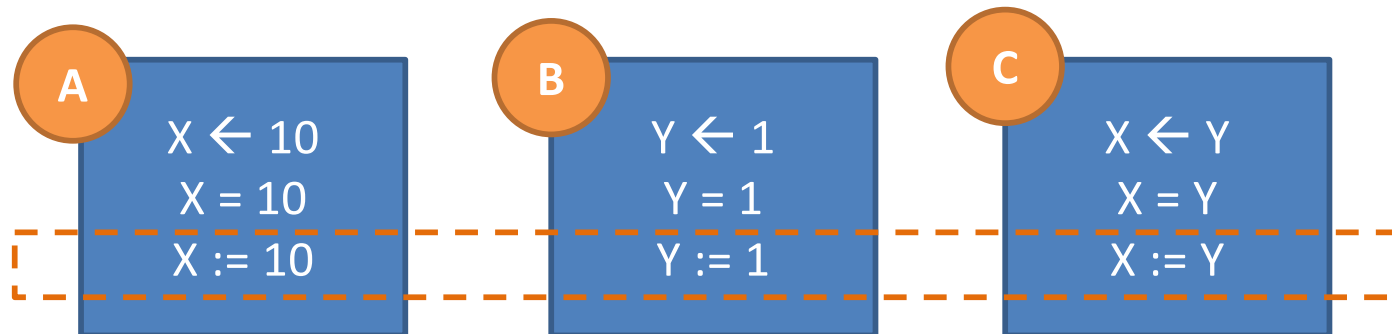
■ Características

- Codificar programas con mayor agilidad.
- Codificar programas en cualquier lenguaje de programación.
- Misma validez semántica.
- Utiliza una serie de frases en lenguaje común, instrucciones de programación, y palabras clave que definen las estructuras básicas.

Pseudocódigo = Pseudo (Supuesto) + Código (Instrucción)

Suponiendo las siguientes instrucciones

- A - “Asigne el valor 10 a X”
- B - “Asigne el valor 1 a Y”
- C - “Reemplace el valor de la variable X por el valor de la variable Y”



Si el valor que se desea asignar es ingresado por teclado, se utiliza la instrucción leer. Si se desea mostrar el valor de una variable en pantalla se utiliza la instrucción escribir:

- Leer (X)
- Escribir (X)

Se utiliza cuando se desea repetir un determinado bloque de instrucciones sobre un índice i (por convención se usa i , sin embargo se puede usar cualquier identificador):

Supongamos las siguientes instrucciones:

- “Sumarle 10 veces a X el valor de Y ”
- “Mostrar en pantalla el valor de la variable X ”

```
X:= 0  
Y:= 1  
para l:=0 hasta 10 hacer  
X:= X+Y  
fin para  
escribir (X)
```

Se utiliza cuando se desea tomar un curso de acción distinto si se verifica una condición.

Supongamos la siguiente instrucción:

- “Leer un número por teclado e imprimir en pantalla si es negativo o positivo”

```
Leer(X)
si X<>0 entonces
    si X>0 entonces
        Escribir(“Es positivo”)
    si no
        Escribir(“Es negativo”)
    fin si
si no
    Escribir(“El número es 0”)
fin si
```


Una construcción similar a la anterior (equivalente en algunos casos)

Supongamos la siguiente instrucción:

- “Leer una edad por teclado e imprimir en pantalla si es niño, adolescente, adulto o mayor”

```
leer (edad)
seleccionar (edad)
  caso edad < 10
    escribir (“Niño”)
  caso (edad > 10 y edad < 21)
    escribir (“Adolescente”)
  caso (edad > 20 y edad < 65)
    escribir (“Adulto”)
  caso edad < 64
    escribir (“Mayor”)
  en otro caso
    escribir (“no esta definido”)
fin seleccionar
```

El bucle se repite mientras la condición sea cierta, si al llegar por primera vez al bucle mientras la condición es falsa, el cuerpo del bucle no se ejecuta ninguna vez.

Dada la siguiente instrucción:

- “Calcular las potencias de números ingresados por teclado siempre que sean positivos e imprimir en pantalla”

```
leer (numero)
mientras numero > 0
    escribir (numero * numero)
    leer (numero)
fin mientras
```

Antes de codificar, leer atentamente el enunciado apuntado en lenguaje natural cuales son las entradas y salidas del problema y qué proceso de transformación se realiza.

1. Calcular el factorial de un numero leído manualmente
2. Calcular el indice de masa corporal de una persona ($IMC = \text{peso (kg)} / \text{talla}^2 \text{ (mts)}^2$)
3. Imprima los números del 0 al 100 en pantalla en orden creciente.
4. Imprimir la suma de los números de 1 a 100 en pantalla.
5. Imprimir los números que hay desde 1 hasta un número ingresado por teclado.
6. Imprimir los números del 0 al 100 en pantalla en orden decreciente.
7. Imprimir la cantidad de palabras ingresadas por teclado hasta ingresar la palabra "fin".
8. Imprimir una frase en pantalla leyendo palabra por palabra hasta ingresar la palabra "fin"

