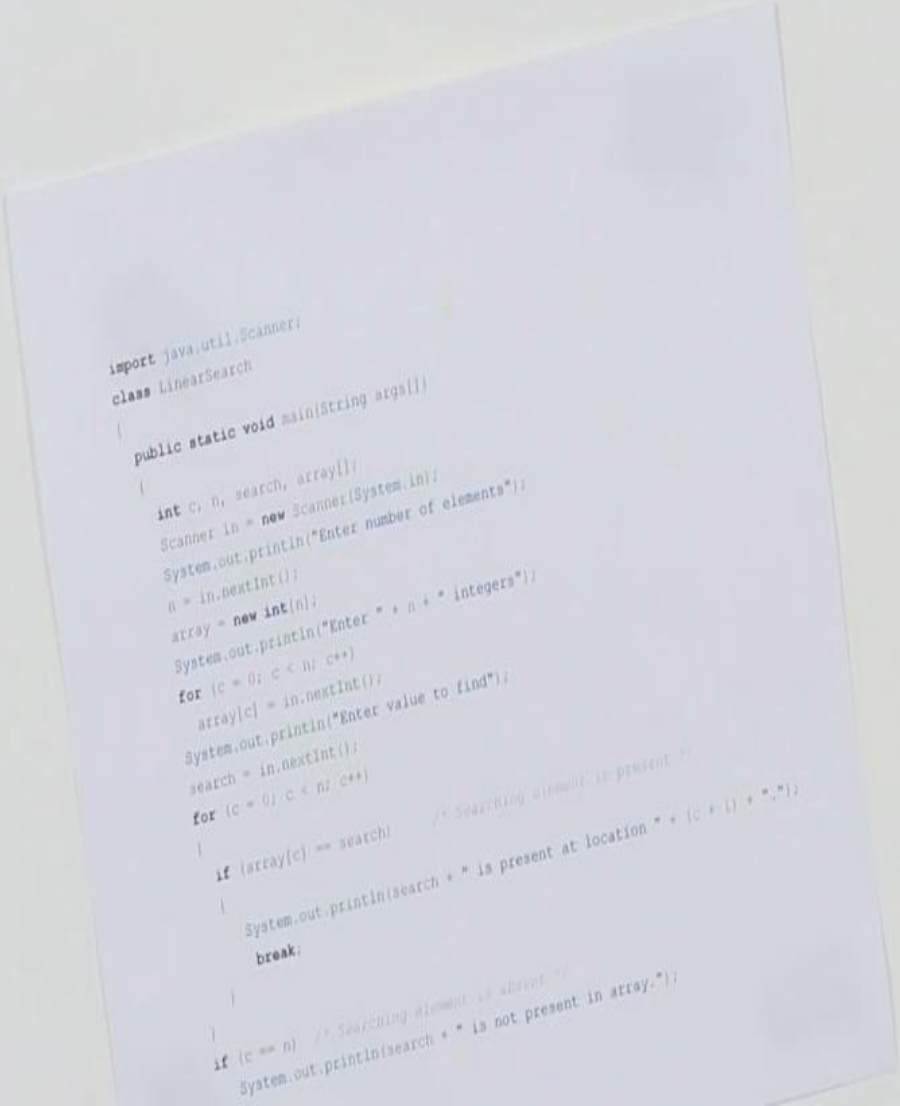
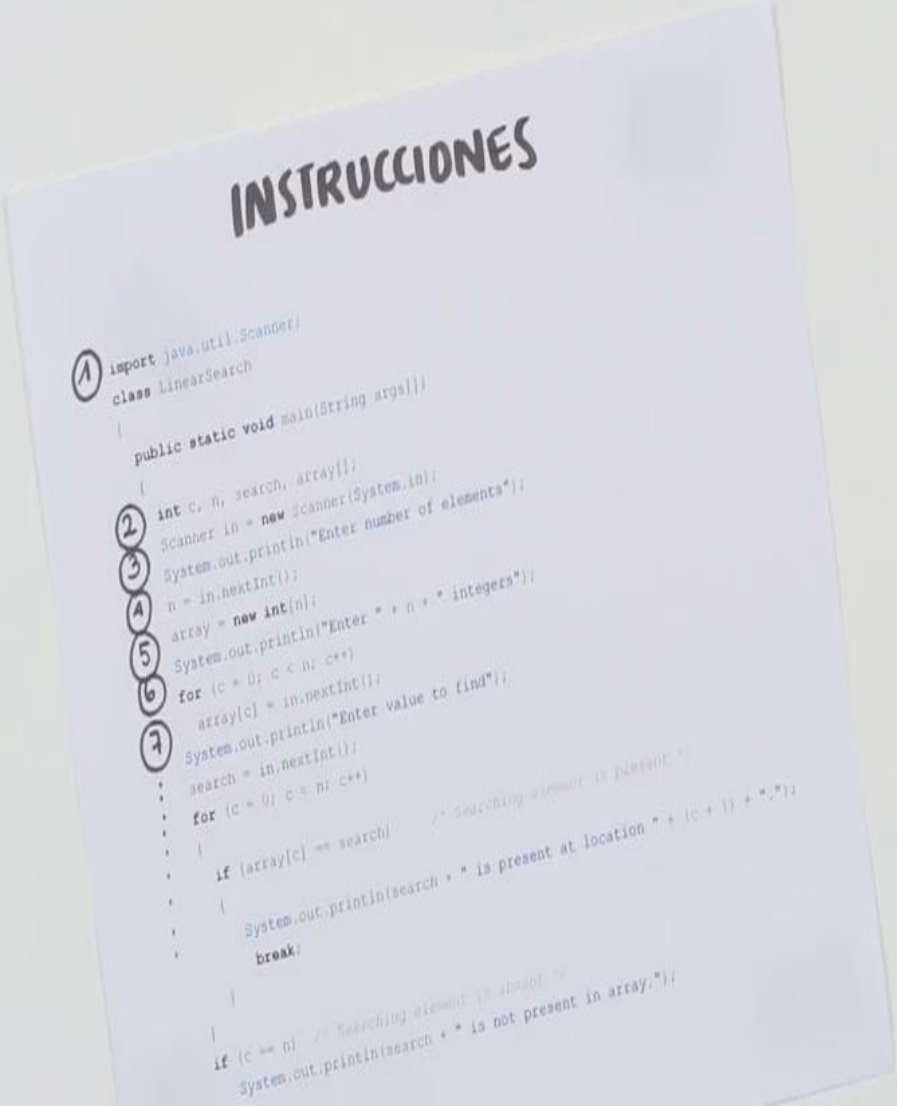
TEMA 3. Programación y Algoritmos de control

ALGORITMOS



```
import java.util.Scanner;
class LinearSearch
{
  public static void main(String args[])
  {
    int c, n, search, array[];
    Scanner in = new Scanner(System.in);
    System.out.println("Enter number of elements");
    n = in.nextInt();
    array = new int[n];
    System.out.println("Enter " + n + " integers");
    for (c = 0; c < n; c++)
      array[c] = in.nextInt();
    System.out.println("Enter value to find");
    search = in.nextInt();
    for (c = 0; c < n; c++)
    {
      if (array[c] == search)     /* Searching element is present */
      {
        System.out.println(search + " is present at location " + (c + 1) + ".");
        break;
      }
    }
    if (c == n)  /* Searching element is absent */
      System.out.println(search + " is not present in array.");
```

ALGORITMOS


INSTRUCCIONES
1 import java.util.Scanner;
class LinearSearch
{
public static void main(String args[])
{
2 int c, n, search, array[];
3 Scanner in = new Scanner(System.in);
4 System.out.println("Enter number of elements");
5 n = in.nextInt();
6 array = new int[n];
7 System.out.println("Enter " + n + " integers");
for (c = 0; c < n; c++)
array[c] = in.nextInt();
System.out.println("Enter value to find");
search = in.nextInt();
for (c = 0; c < n; c++)
{
if (array[c] == search)
{
System.out.println(search + " is present at location " + (c + 1) + ".");
break;
}
}
if (c == n)
System.out.println(search + " is not present in array.");

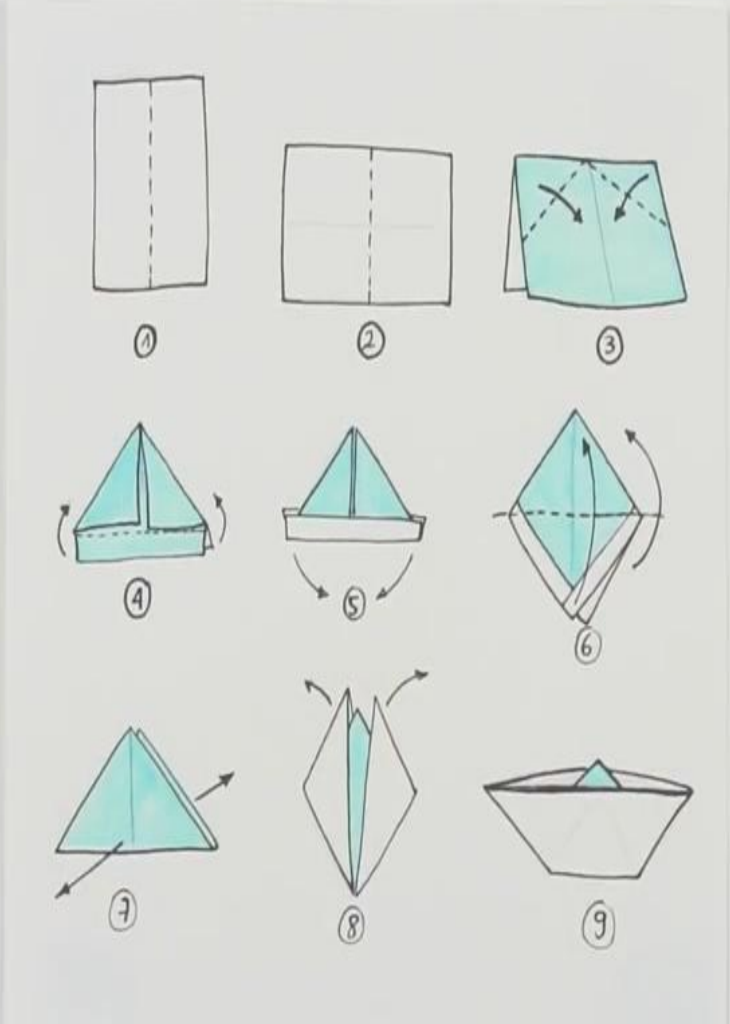
ALGORITMOS

PARA LO QUE SEA

ALGORITMOS
PARA LO QUE SEA
INPUT

ALGORITMOS

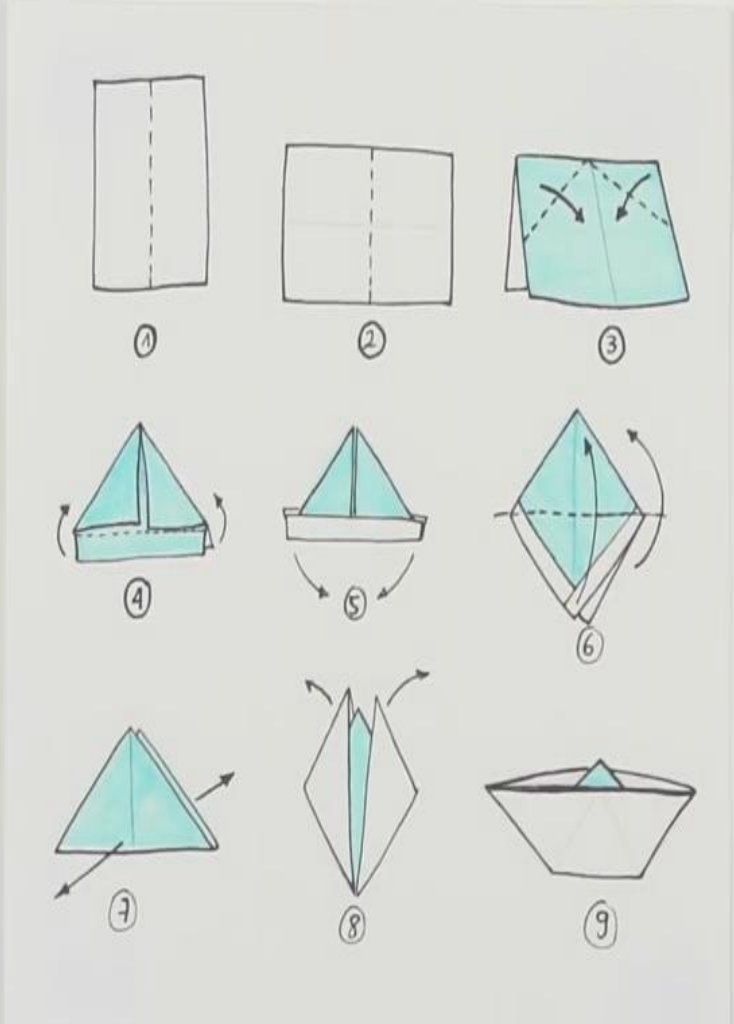
PARA LO QUE SEA

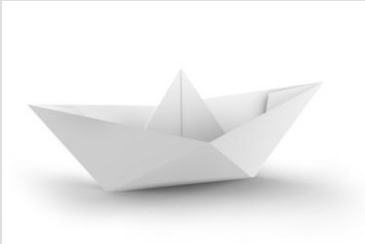


INPUT

ALGORITMOS

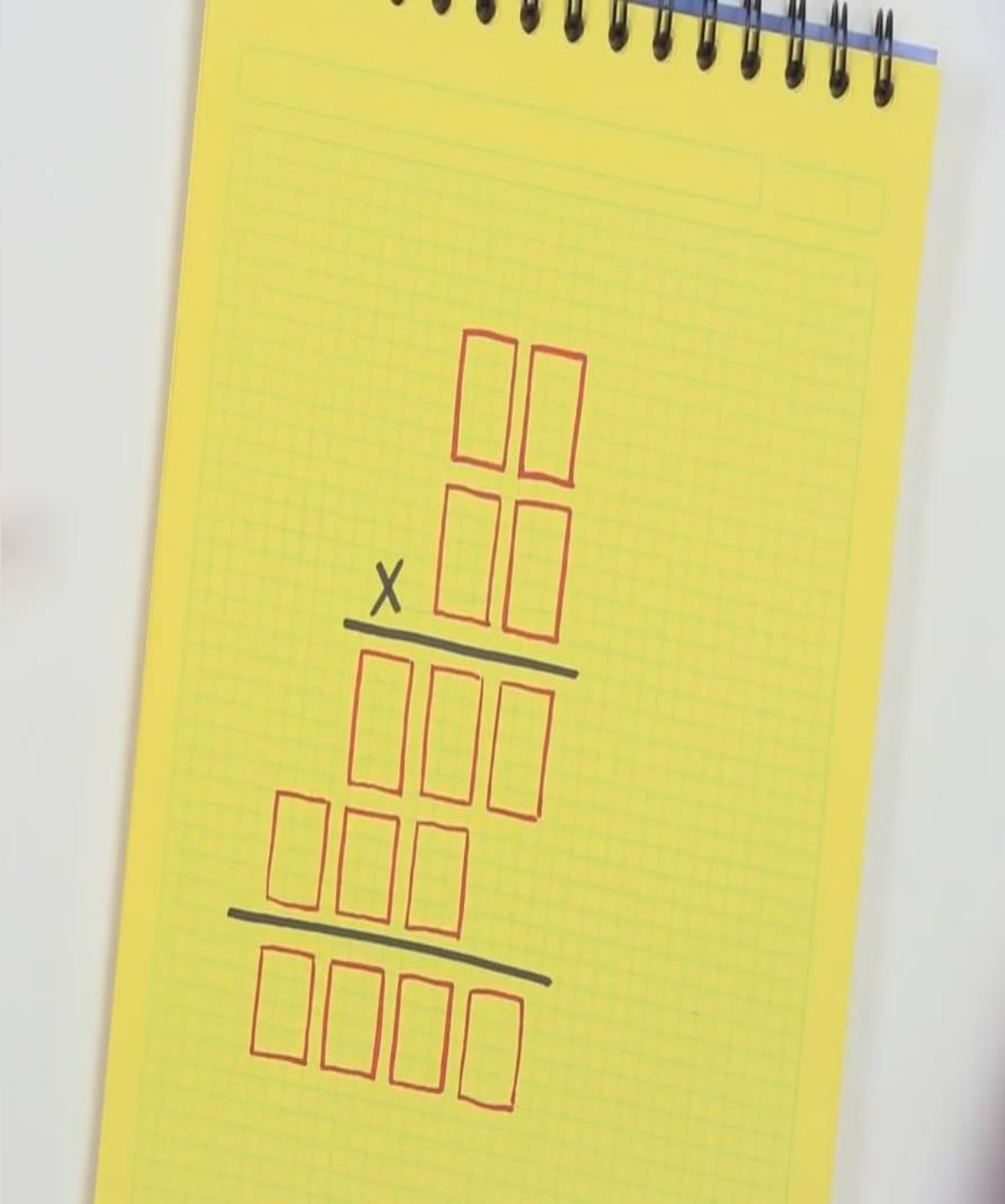
PARA LO QUE SEA

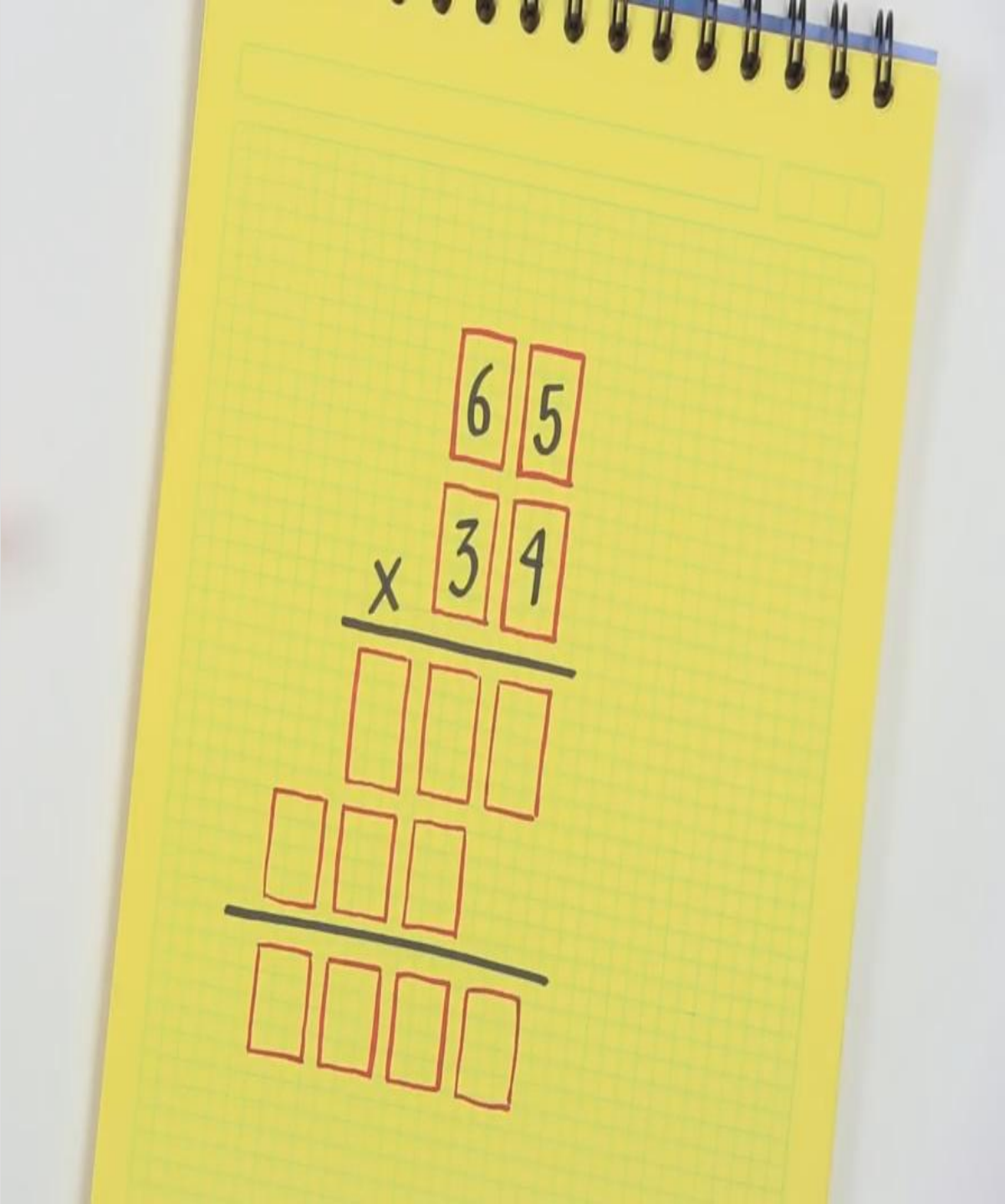


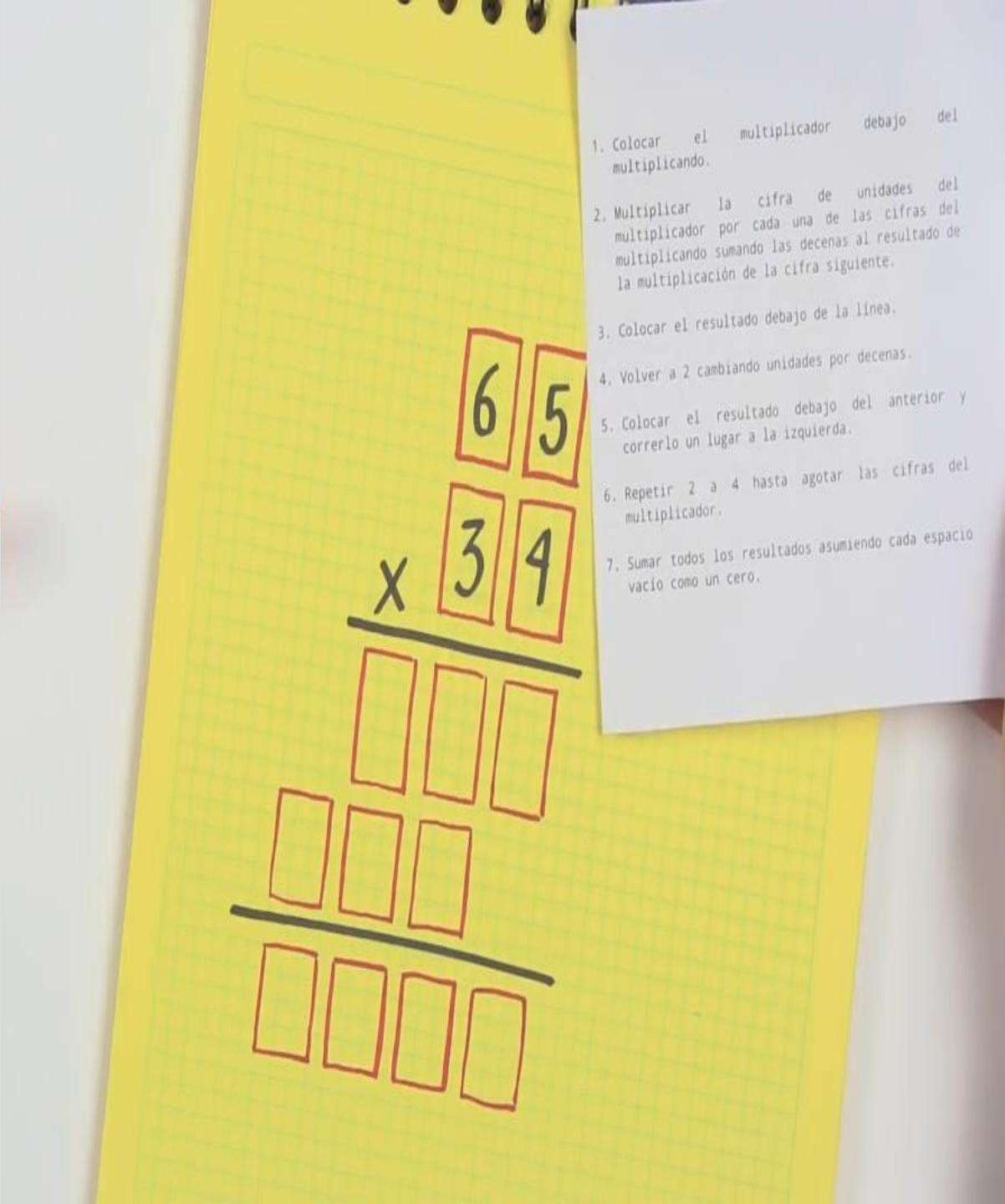


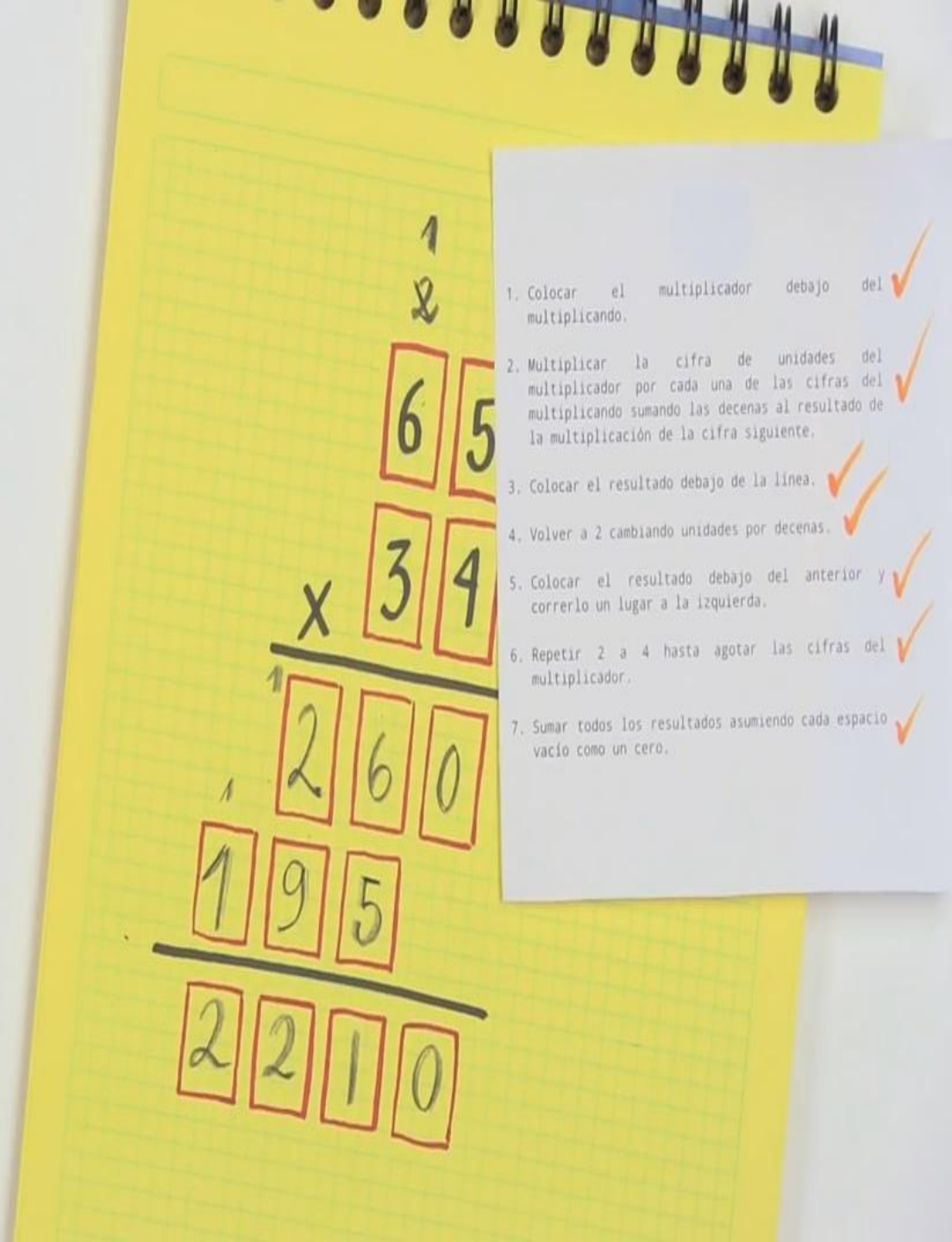
INPUT

OUTPUT


X

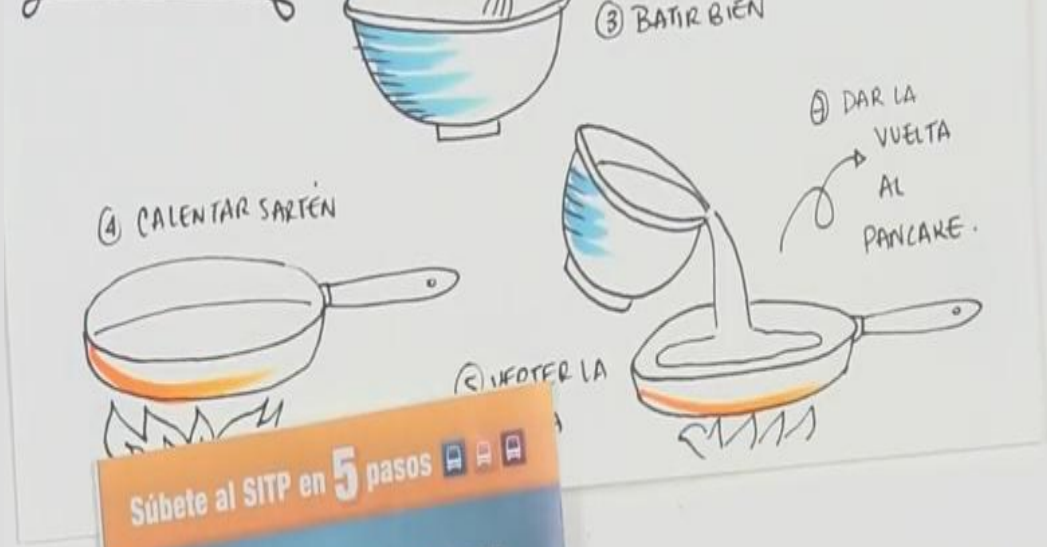

65
x 34

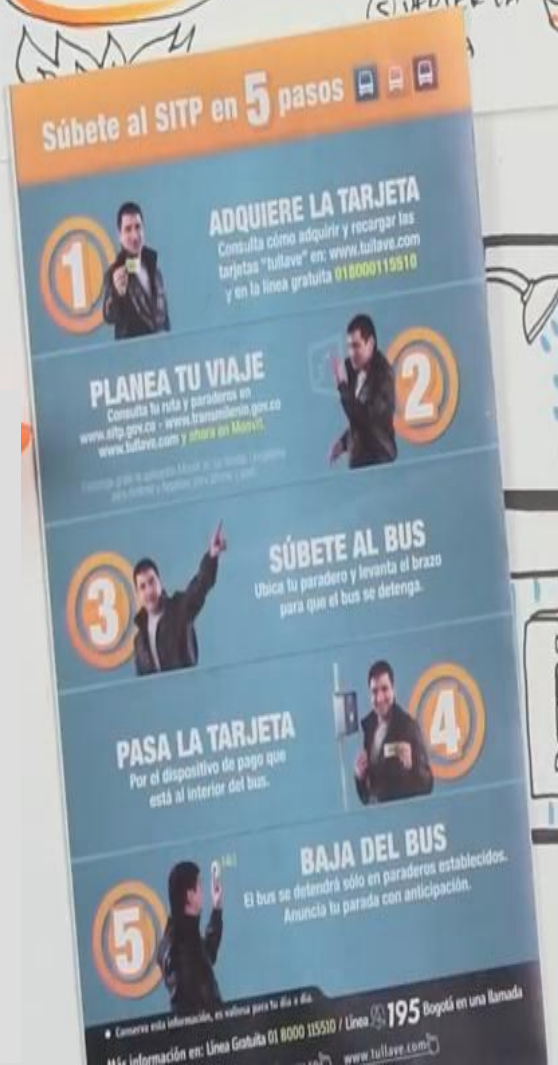

1. Colocar el multiplicador debajo del multiplicando.
2. Multiplicar la cifra de unidades del multiplicador por cada una de las cifras del multiplicando sumando las decenas al resultado de la multiplicación de la cifra siguiente.
3. Colocar el resultado debajo de la línea.
4. Volver a 2 cambiando unidades por decenas.
5. Colocar el resultado debajo del anterior y correrlo un lugar a la izquierda.
6. Repetir 2 a 4 hasta agotar las cifras del multiplicador.
7. Sumar todos los resultados asumiendo cada espacio vacío como un cero.
65
x 34


1. Colocar el multiplicador debajo del multiplicando.
2. Multiplicar la cifra de unidades del multiplicador por cada una de las cifras del multiplicando sumando las decenas al resultado de la multiplicación de la cifra siguiente.
3. Colocar el resultado debajo de la línea.
4. Volver a 2 cambiando unidades por decenas.
5. Colocar el resultado debajo del anterior y correrlo un lugar a la izquierda.
6. Repetir 2 a 4 hasta agotar las cifras del multiplicador.
7. Sumar todos los resultados asumiendo cada espacio vacío como un cero.
65
x 34
260
195
2210

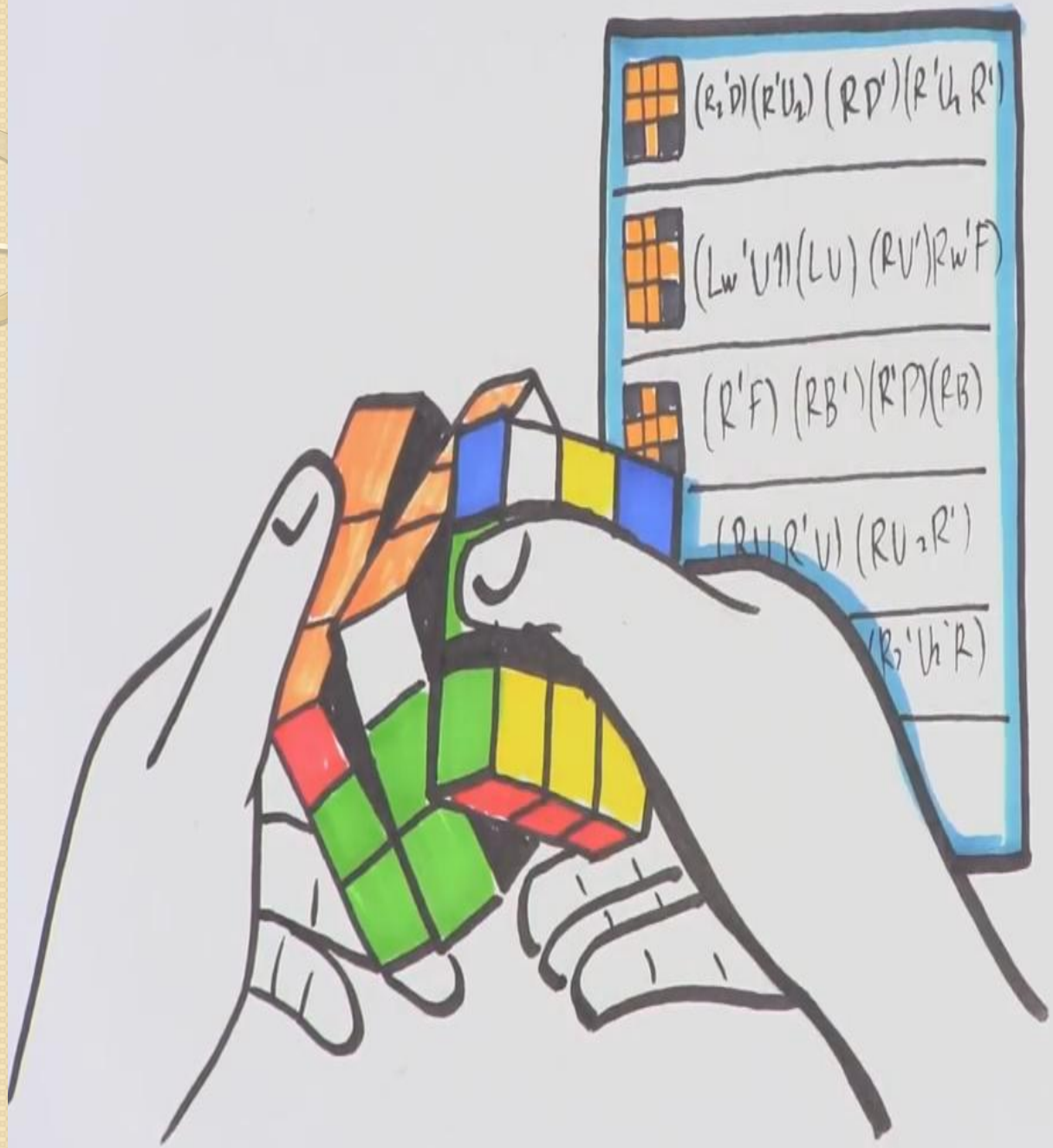
Pancakes
1 TAZA DE HARINA
2 TAZAS DE LECHE
3 CUCHARADAS DE MANTEQUILLA
3 HUEVOS
(CANELA Y VAINILLA OPCIONALES)
① derretir mantequilla
HARINA
② MEZCLAR TODO
③ BATIR BIEN
④ CALENTAR SARTÉN
⑤ VERTER LA MASA
⑥ DAR LA VUELTA AL PANCAKE.

Pancakes
1 TAZA DE HARINA
2 TAZAS DE LECHE
3 CUCHARADAS DE MANTEQUILLA
3 HUEVOS
(CANELA Y VAINILLA OPCIONALES)
① derretir mantequilla
HARINA
② MEZCLAR TODO
③ BATIR BIEN
④ CALENTAR SARTÉN
⑤ VERTER LA MASA
⑥ DAR LA VUELTA AL PANCAKE.


③ BATIR BIEN
④ CALENTAR SARTÉN
⑤ VOLTEAR LA
⑥ DAR LA VUELTA AL PANCAKE.

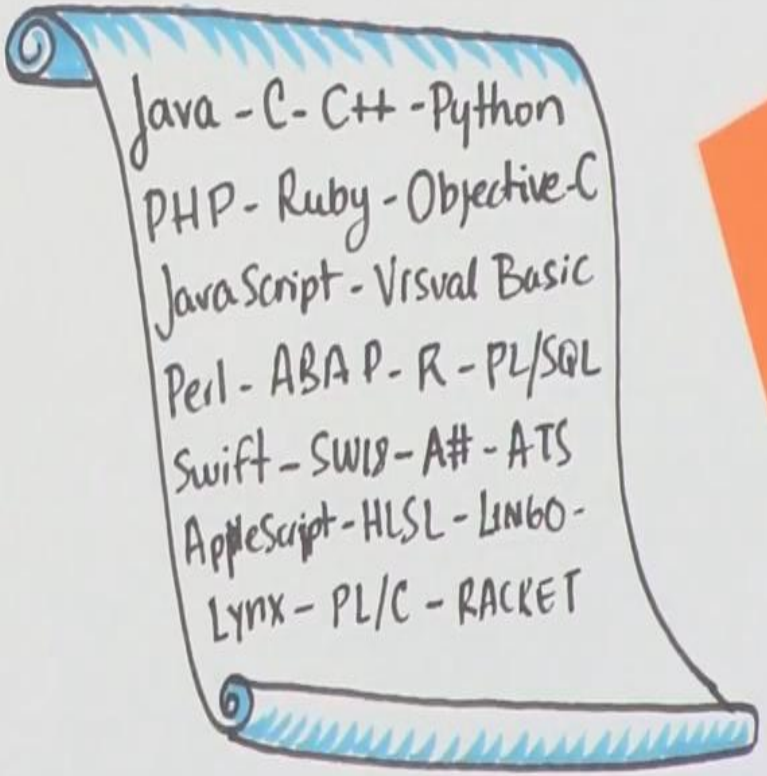

Súbete al SITP en 5 pasos
1
ADQUIERE LA TARJETA
Consulta cómo adquirir y recargar las tarjetas "tullave" en: www.tullave.com y en la línea gratuita 018000115510
2
PLANEA TU VIAJE
Consulta tu ruta y paraderos en www.sitp.gov.co - www.transmilenio.gov.co www.tullave.com y ahora en Moovit.
3
SÚBETE AL BUS
Ubica tu paradero y levanta el brazo para que el bus se detenga.
4
PASA LA TARJETA
Por el dispositivo de pago que está al interior del bus.
5
BAJA DEL BUS
El bus se detendrá sólo en paraderos establecidos. Anuncia tu parada con anticipación.
Más información en: Línea Gratuita 01 8000 115510 / Línea 195 Bogotá en una llamada
www.tullave.com

ON
OFF
ON
OFF


(R2'D)(R'U2) (RD')(R'U2 R')
(Lw'U')(LU) (RU')Rw'F)
(R'F) (RB')(R'F)(RB)
(RU2 R'U) (RU2R')
(R2'U2'R)

Programación: (pro.ɣrama 'θjon)

1. El arte de convertir las cosas que hacemos o las que quisiéramos hacer en una secuencia de instrucciones que un computador pueda prousar, en un algoritmo



CARACTERÍSTICAS de un ALGORITMO

Debe ser FINITO

Debe ser EFICIENTE

Debe ser LEGIBLE

Debe ser MODIFICABLE

Debe ser MODULAR

Debe ser PRECISO

Debe tener UN UNICO PUNTO DE ENTRADA, PROCESO Y PUNTO DE SALIDA

PROGRAMA o ALGORITMO

NO Confundir ALGORITMO con PROGRAMA INFORMÁTICO.

ALGORITMO: Lista de pasos organizados que establecen el proceso que se debe seguir para obtener la solución de un problema específico.

PROGRAMA: Conjunto de algoritmos.

ELABORACIÓN DE UN ALGORITMO **"Mayor de dos números"**

USANDO UNA DESCRIPCIÓN DE ALTO NIVEL

Paso 1: Introduce el primer número

Paso 2: Introduce el segundo número

Paso 3: Si el primer número es mayor que el segundo número entonces decir que el número mayor es el primero

Paso 4: Si el segundo número es mayor que el primer número entonces decir que el número mayor es el segundo

Paso 5: Si el primer número es igual que el segundo número entonces decir que los dos números son iguales

ELABORACIÓN DE UN ALGORITMO

USANDO UNA DESCRIPCIÓN FORMAL (PSUDOCÓDIGO)

```
Inicio
        Pedir el número A
        Pedir el número B
        Si A > B entonces decir "El mayor es A"
        Si B > A entonces decir "El mayor es B"
        Si A = B entonces decir "Son iguales"
Fin
```

ELABORACIÓN DE UN ALGORITMO

IMPLEMENTANDO EN UN LENGUAJE DE PROGRAMACIÓN (ejemplo en C)

```
Mayor ()
{
        printf("Introduce el número A");
        scanf(A);
        printf("Introduce el número B");
        scanf(B);
        if (A > B) printf( "El mayor es A");
        if (B > A) printf( "El mayor es B");
        if (A == B) printf( "A y B son iguales");
}
```

ELABORACIÓN DE UN ALGORITMO

USANDO UNA DESCRIPCIÓN DE ALTO NIVEL

Paso 1: Introduce el primer número

Paso 2: Introduce el segundo número

Paso 3: Si el primer número es mayor que el segundo número entonces decir que el número mayor es el primero

Paso 4: Si el segundo número es mayor que el primer número entonces decir queel número mayor es el segundo

Paso 5: Si el primer número es igual que el segundo número entonces decir que los dos números son iguales

USANDO UNA DESCRIPCIÓN FORMAL (PSUDOCÓDIGO)

```
Inicio
    Pedir el número A
    Pedir el número B
    Si A > B entonces decir "El
            mayor es A"
    Si B > A entonces decir "El
            mayor es B"
    Si A = B entonces decir "Son
            iguales"
Fin
```

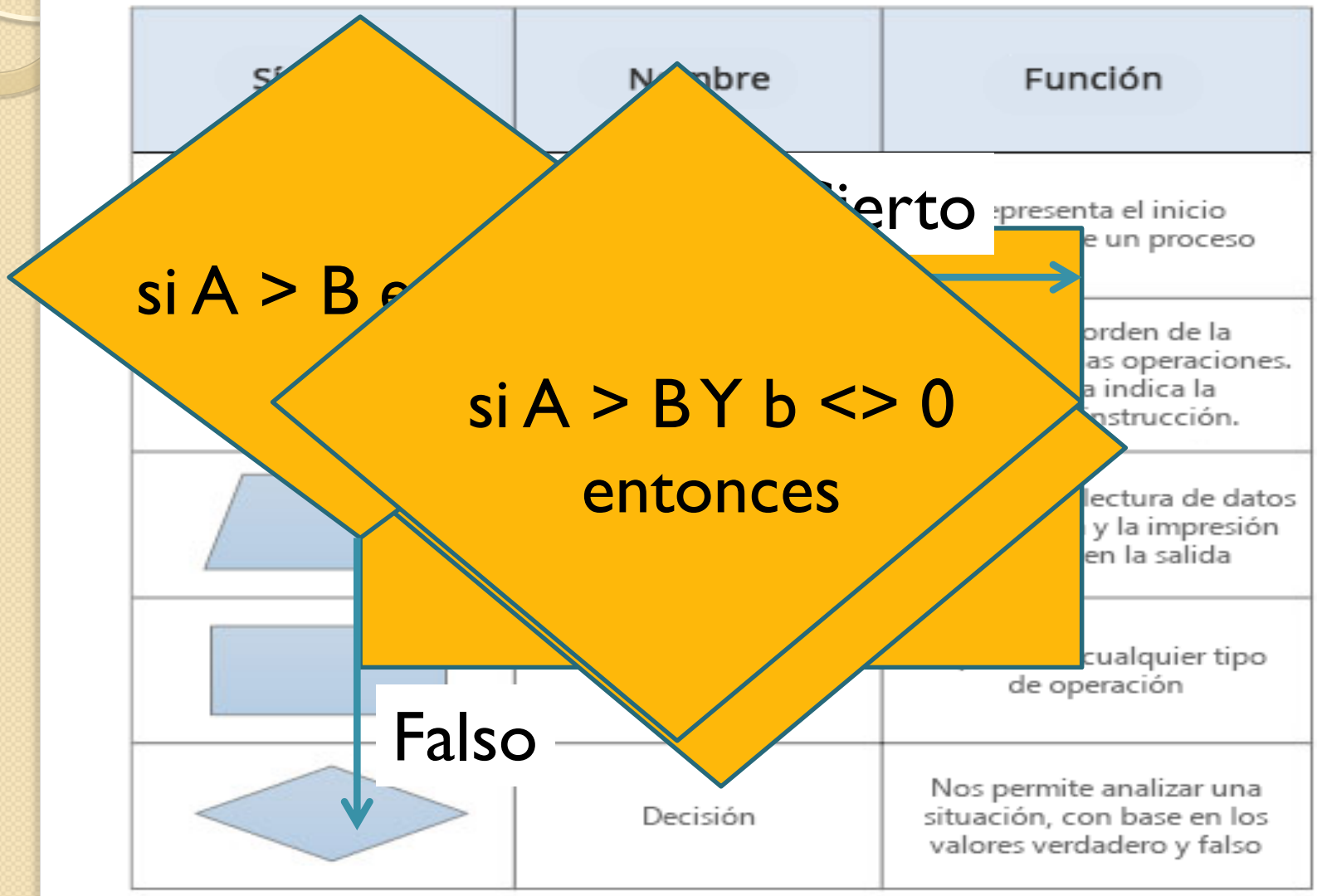
IMPLEMENTANDO EN UN LENGUAJE DE PROGRAMACIÓN (ejemplo en C)

```
Mayor ()
{
  printf("Introduce el número
            A");
  scanf(A);
  printf("Introduce el número
            B");
  scanf(B);
  if (A > B) printf( "El mayor
            es A");
  if (B > A) printf( "El mayor
            es B");
  if (A == B) printf( "A y B son
            iguales");
}
```

Representación de un algoritmo: Diagramas de flujo

- Representar de forma gráfica un proceso, como un algoritmo
- Se empieza en un punto de INICIO y el objetivo es llegar a un punto FINAL
- Se usan símbolos y flechas para indicar cada paso y el orden
- Es útil para pequeños algoritmos, si es muy grande, la complejidad gráfica llega a ser contraproducente
- Existen varios tipos de diagramas de flujo, el mas usado (el nuestro), es de tipo VERTICAL, donde el flujo va de arriba hacia abajo.

Representación de un algoritmo: Diagramas de flujo



Representación de un algoritmo: Diagramas de flujo

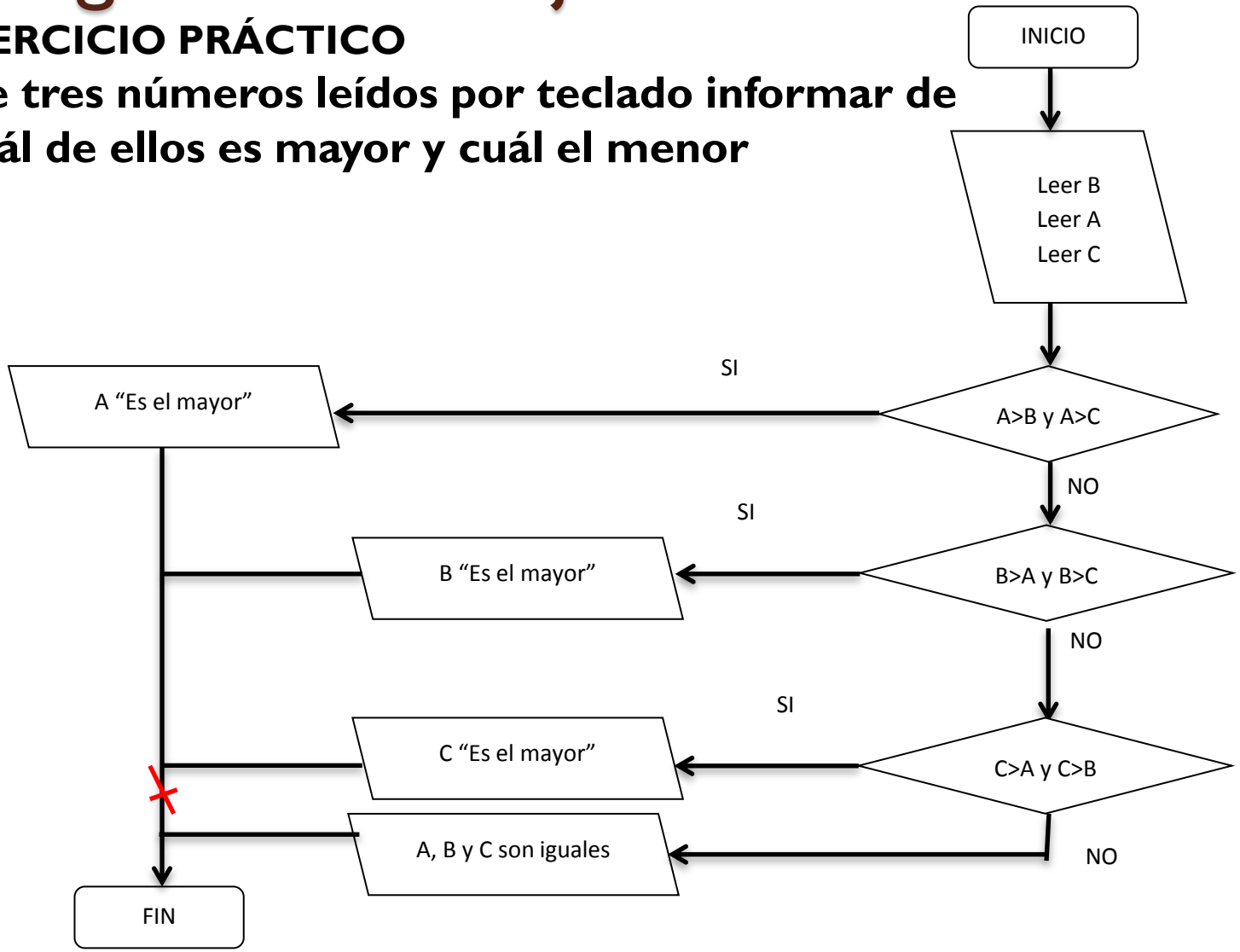
EJERCICIO PRÁCTICO

De tres números leídos por teclado informar de cuál de ellos es mayor y cuál el menor

Representación de un algoritmo: Diagramas de flujo

EJERCICIO PRÁCTICO

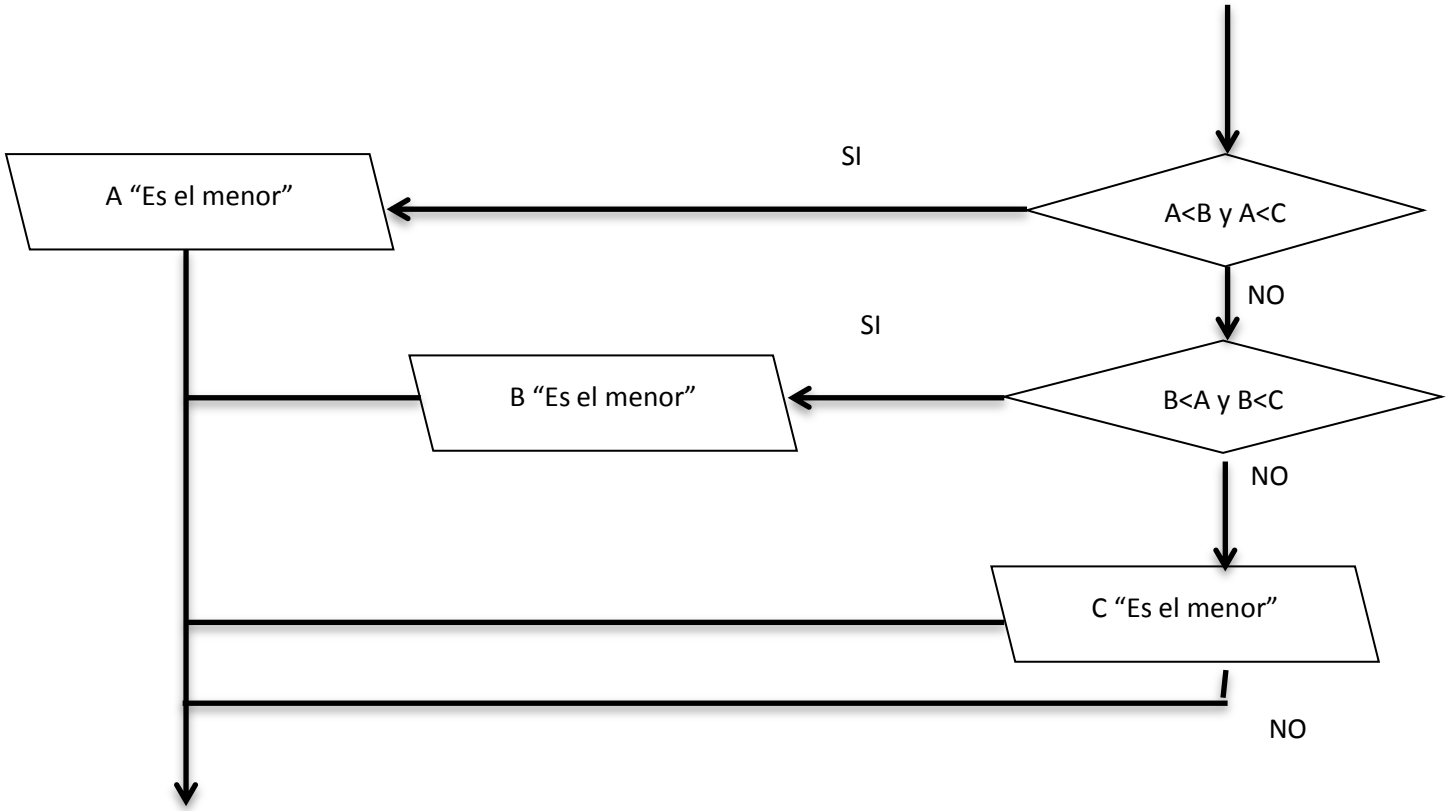
De tres números leídos por teclado informar de cuál de ellos es mayor y cuál el menor



Representación de un algoritmo: Diagramas de flujo

EJERCICIO PRÁCTICO

De tres números leídos por teclado informar de cuál de ellos es mayor y cuál el menor



Hacer un huevo frito

1. Si no tengo sartén ir a fin
2. Si no tengo huevo ir a fin
3. Si no tengo aceite ir a fin
4. Si no funciona fuego ir a fin
5. Encender fuego
6. Poner sartén en el fuego
7. Echar aceite en la sartén
8. Si aceite no está caliente entonces ir a paso 8
9. Romper huevo sobre sartén
10. Si tengo sal echar sal sobre el huevo
11. Si el huevo aun no está frito entonces ir al paso 11
12. Poner huevo en un plato

Averiguar la edad

Diseñar un algoritmo en Alto Nivel, pseudocódigo y diagrama de flujo para

Que pidiendo el año de nacimiento y el año actual, decir la edad. Hay que comprobar que el año actual sea mayor que el año de nacimiento.

Averiguar la edad

Alto Nivel

Paso 1 Pedir el año de nacimiento

Paso 2 Pedir el año actual

Paso 3 Si el año actual no es menor o igual al año de nacimiento entonces ir al paso 2

Paso 4 La edad será igual al año actual menos el año de nacimiento

Paso 5 Decir la edad

Averiguar la edad

Pseudocódigo

```
Inicio
  Decir “Introduce el año de nacimiento”
  Pedir ANACIMIENTO
  Decir “Introduce el año actual”
  Pedir AACTUAL
  Si AACTUAL <= ANACIMIENTO entonces
            volver a pedir AACTUAL
  EDAD = AACTUAL – ANACIMIENTO
  Decir “Tu edad es EDAD años”
Fin
```

Averiguar la edad

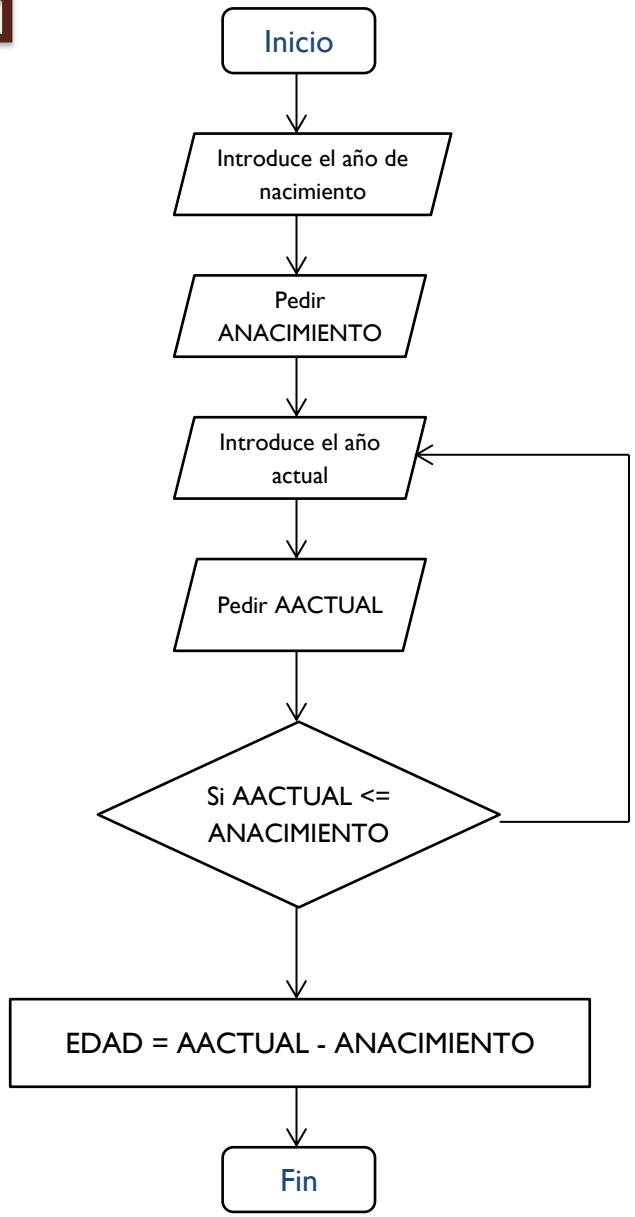



Diagrama de flujo