

INFORMÁTICA

ALGORITMIA Y
PROGRAMACIÓN

COCINAR UN HUEVO DURO



INTRODUCCIÓN

- Como ya sabemos, el ordenador por sí solo no es capaz de resolver ningún problema, es necesario describir, de forma detallada y entendible para él, todas aquellas operaciones necesarias para efectuar la resolución de un problema → **programa.**
- Para ser un programador eficaz es necesario aprender a resolver problemas de un modo riguroso y sistemático, lo que se denomina **metodología de la programación.**
- El proceso de programación deber ser independiente del lenguaje que se vaya a utilizar.

INTRODUCCIÓN

- El proceso que hay que seguir para llegar a la resolución de un problema se podría dividir en tres fases:
 - **Fase de análisis:** definición y comprensión clara del problema para que pueda ser analizado con todo detalle.
 - **Fase de programación:** elaboración del método paso a paso para solucionar el problema.
 - **Fase de codificación:** traducción del algoritmo a un lenguaje de programación y validación de que se ha realizado adecuadamente.
- Una vez terminadas estas tres fases básicas se realiza la instalación y puesta a punto de los programas
- Para ello, se usan conjuntos de datos también denominados baterías de pruebas que sirven para determinar si el programa o programas funcionan correctamente

INTRODUCCIÓN

- En caso de que los programas no funcionen adecuadamente será necesaria su **depuración**.
- Una vez depurados se procede a realizar y completar la documentación.
Tipos:
 - Documentación interna:
 - Es la que se encuentra dentro del texto o código del propio programa.
 - Normalmente se realiza durante la fase de codificación.
 - Facilita las modificaciones posteriores de los programas, ya que probablemente serán realizadas por personas distintas o por la misma pero después de un largo período de tiempo.
 - Documentación externa:
 - Formada por un conjunto de documentos en los que se indica los algoritmos que se han utilizado, las estructuras de datos, las entradas posibles, el comportamiento del sistema, los requisitos hardware, etc.
 - Entre estos documentos cabe destacar el **manual del programador** y el **manual de usuario**.

FASE DE ANÁLISIS

- **Objetivo:** ayudar al programador a alcanzar una cierta comprensión de la naturaleza del problema.
- Para ello, es muy importante que el problema esté **bien definido.**
- La correcta definición de un problema requiere una **descripción detallada de las especificaciones de entrada y salida.**

ANALISIS DEL PROBLEMA

El análisis del problema requiere:

- Obtener una idea general de cuál es el problema.
- Determinar qué información debe proporcionar la resolución del problema (salidas) y qué datos iniciales necesita (entradas).
- Identificar procedimientos generales que puedan llevar de las entradas a las salidas.
 - Datos de entrada
 - Procedimiento
 - Datos de salida deseados

Cuando un problema es complejo hay que dividirlo en **subproblemas**.

FASE DE PROGRAMACIÓN

- Un ordenador no tiene capacidad para resolver un problema a menos que se le proporcione la solución como una **secuencia de pasos a realizar**.
- Estos pasos representan las instrucciones que la máquina deberá ejecutar y constituyen lo que se denomina el **algoritmo**.
- Esta fase consiste **en diseñar una solución en forma de algoritmo para el problema planteado**.
- Los algoritmos son independientes tanto del lenguaje de programación en el que se expresen posteriormente como el computador donde se ejecuten.
- En la programación es más importante el algoritmo que el lenguaje de programación.

ALGORITMIA

Algoritmo:

Procedimiento paso a paso para solucionar un problema.

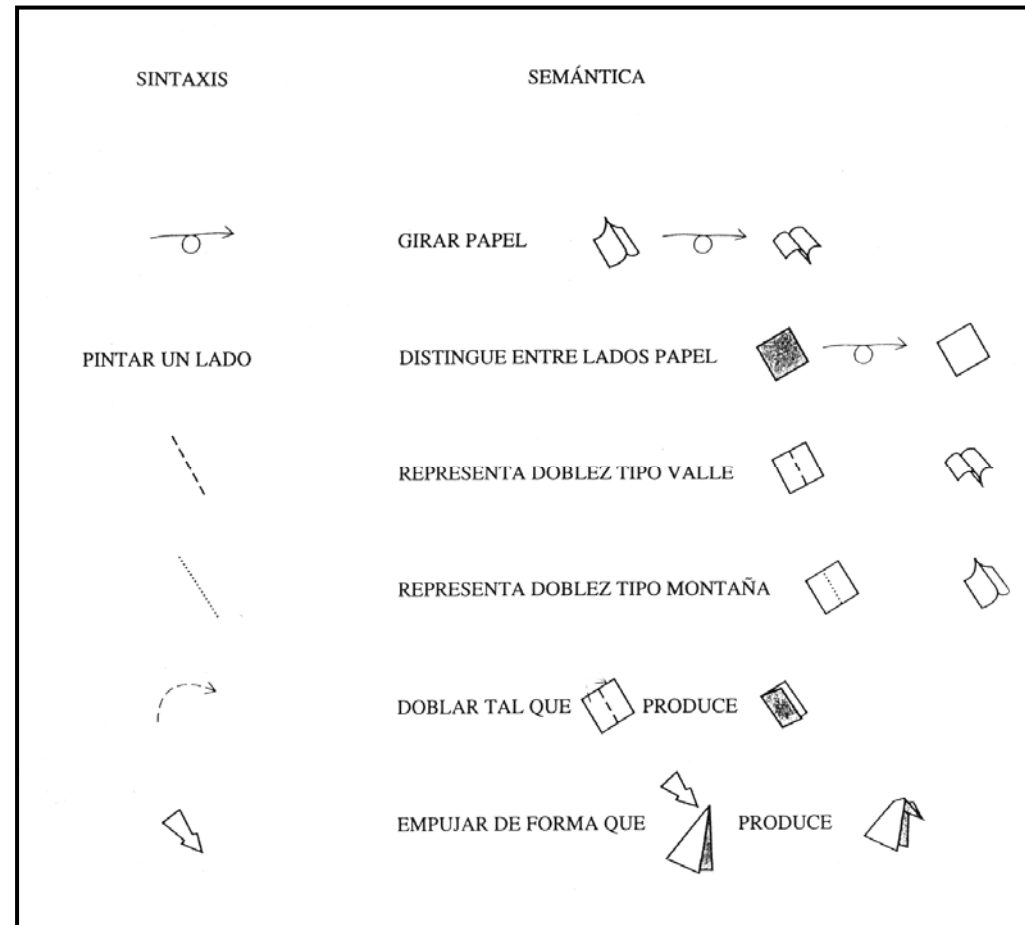
FASE DE PROGRAMACIÓN

Ejemplo: descripción de un algoritmo, mediante lenguaje natural, para ir al cine a ver “BATMAN”

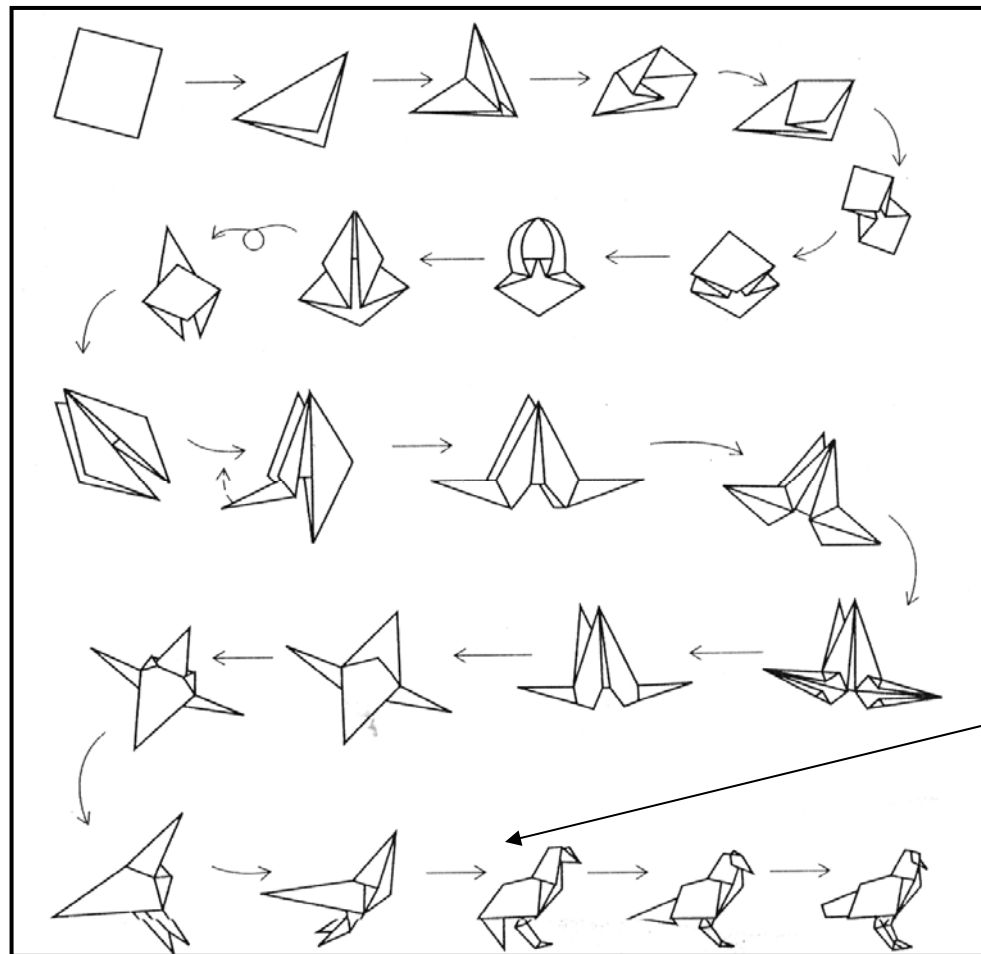
```
Inicio
Ver la cartelera de cines en el periódico
Si proyectan BATMAN entonces
    Ir al cine
    Si hay cola entonces
        Ponerse en ella
    Mientras haya personas delante hacer
        Avanzar en la cola
    Si existen asientos disponibles entonces
        Comprar una entrada
        Encontrar el asiento correspondiente
        Mientras proyectan la película
            Ver la película
    Sino (existen asientos disponibles)
        Abandonar el cine
        Volver a casa
    Sino (proyectan BATMAN)
        Decidir otra actividad
Fin
```

ALGORITMOS

- Se define una **sintáxis**
- Se establece correspondencia con una **semántica**.

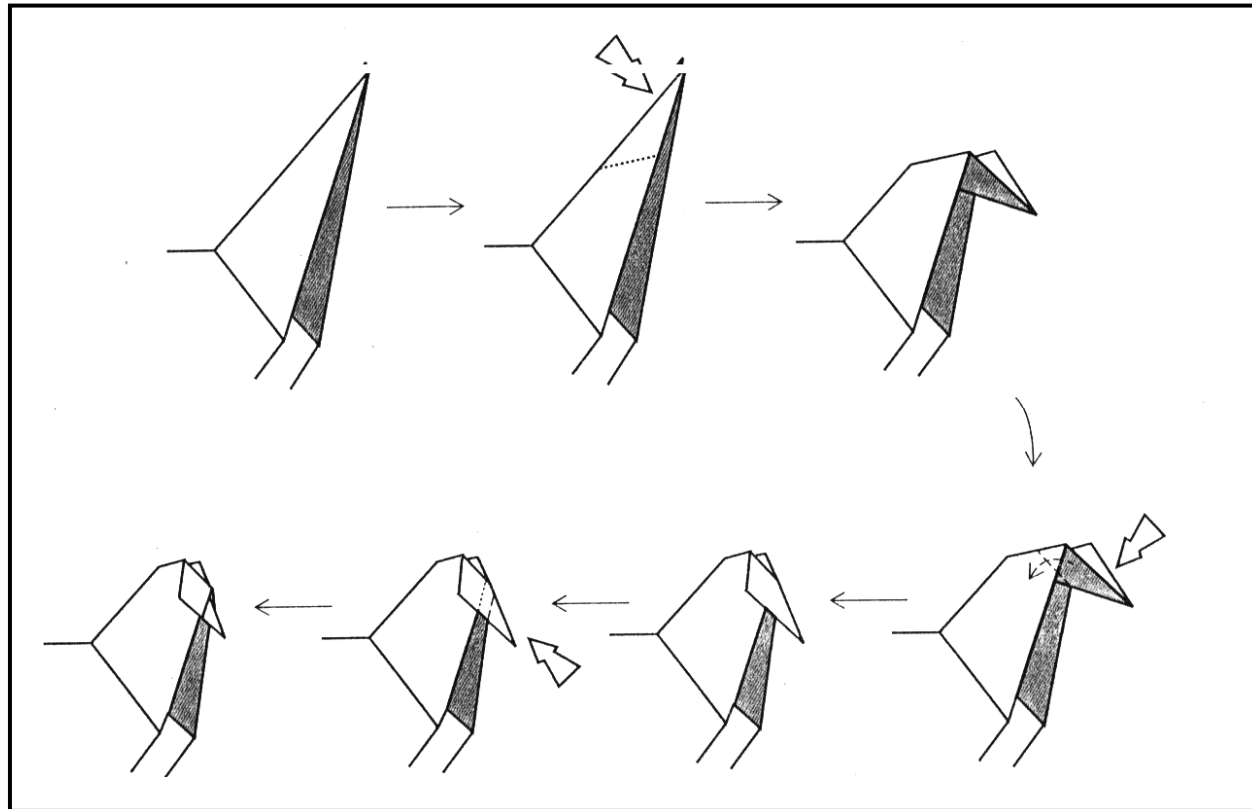


PRIMER ALGORITMO, HACER UNA PAJARITA DE PAPEL



Subrutina

SUBROUTINA



CARACTERÍSTICAS DE UN ALGORITMO

Las características fundamentales que debe cumplir todo algoritmo son:

- Debe ser **preciso** e indicar el orden de realización de cada paso.
- Debe estar **definido** (determinista): si se siguen los mismos pasos de un algoritmo n veces se debe obtener siempre el mismo resultado.
- Debe ser **finito**: si se siguen los pasos de un algoritmo se debe terminar en algún momento, es decir, tiene un número finito de pasos.

Para la descripción de un algoritmo se usan distintas representaciones como el lenguaje natural, el pseudocódigo, los árboles de decisión o los organigramas.

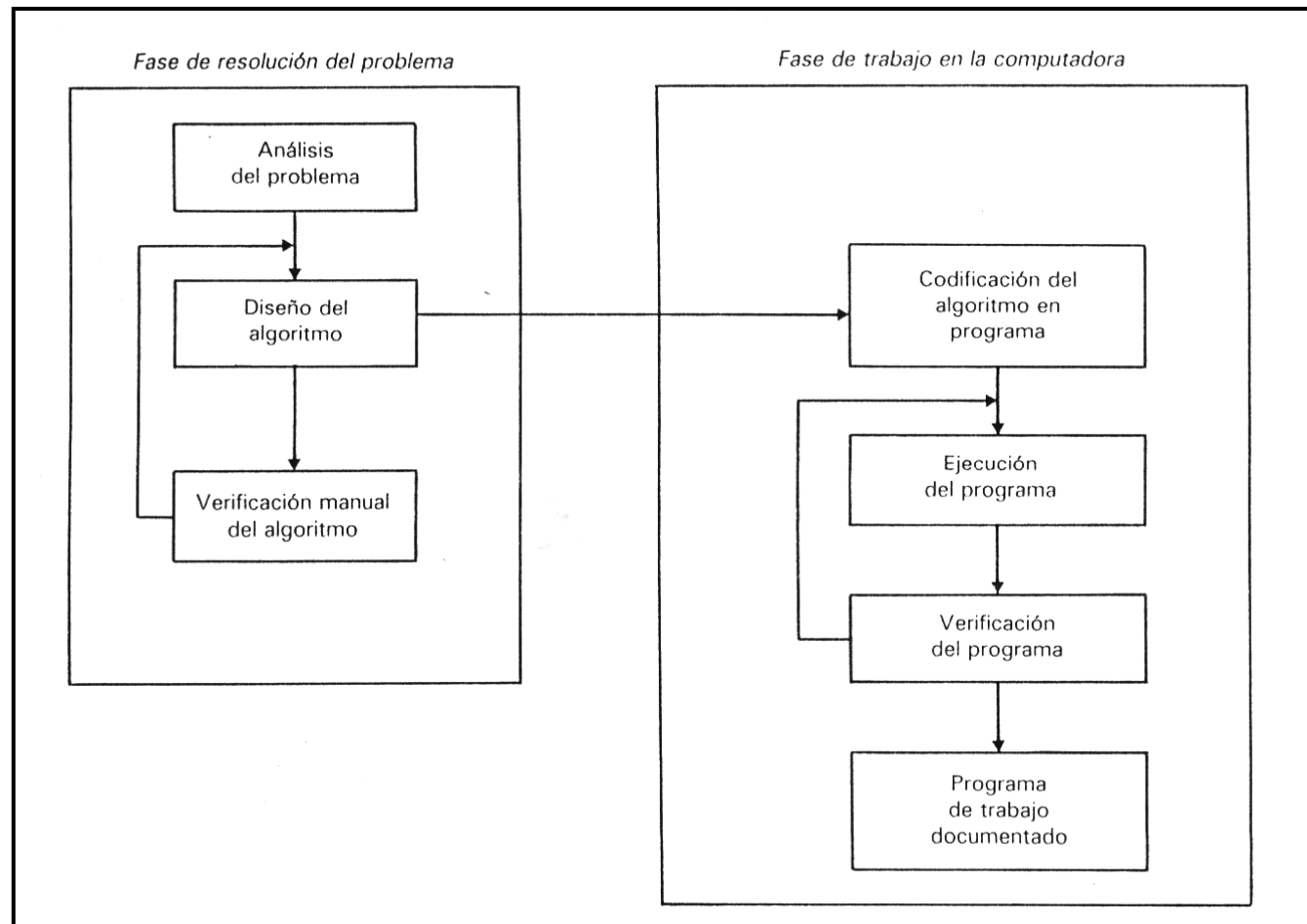
FASE DE CODIFICACIÓN

Una vez diseñado el algoritmo se pasa a la fase de resolución práctica del algoritmo en el computador.

Esta fase puede descomponerse en 3 subfases:

- **Implementación** del algoritmo: conversión del algoritmo en algún lenguaje de programación. El algoritmo escrito en el lenguaje de programación se denomina **código** o **programa**.
- **Verificación** del programa: consiste en comprobar que el programa esté correctamente escrito siguiendo las normas léxicas y sintácticas del lenguaje.
- **Validación** del programa: consiste en comprobar que los resultados proporcionados por el programa se corresponden con los especificados en la definición del problema.

PROGRAMAR

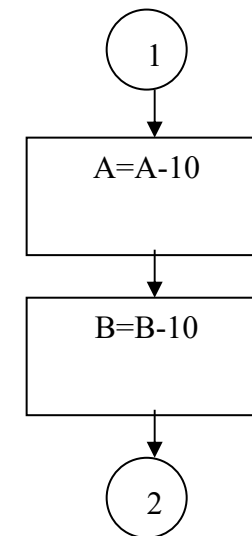
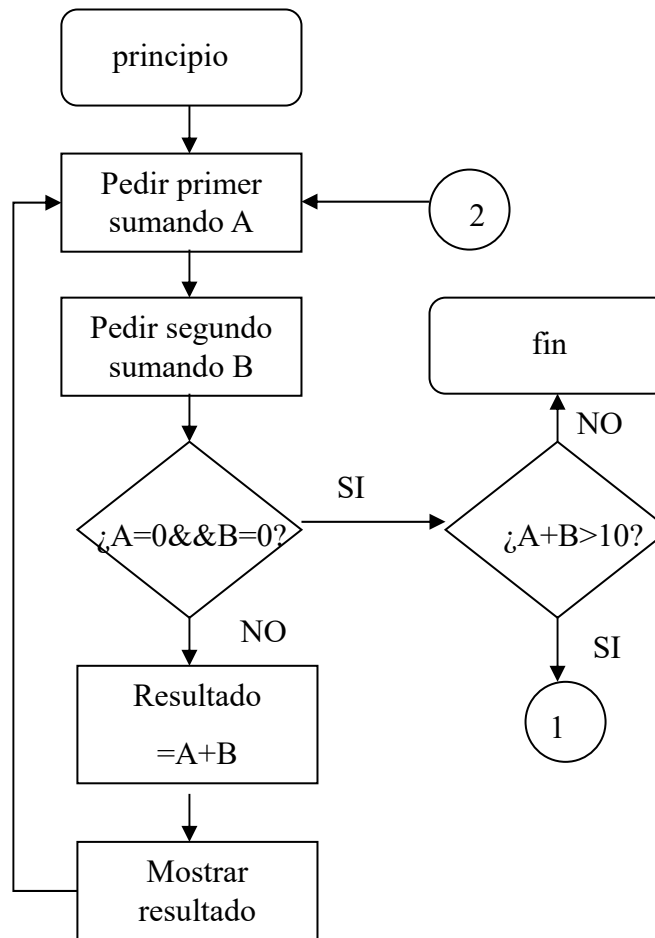


PROGRAMA

- Conjunto de instrucciones que se ejecutan en un ordenador:
 - Secuencialmente.
 - En paralelo.
- El ordenador maneja las instrucciones y datos representados como 0s o 1s (lenguaje máquina).
- Nosotros programamos un ordenador en lenguajes de más alto nivel que el máquina por comodidad, pero se requiere uno o más pasos para que el ordenador pueda utilizar nuestro programa.

PREPARACIÓN DE UN PROGRAMA

Diagramas de flujo.



PREPARACIÓN DE UN PROGRAMA

Pseudo código

Pedir primer dato A

Pedir segundo dato B

Si $A=0 \& \& B=0$

 Si $A+B > 10$

 ir a subrutina 1

 Else terminar

 resultado = $A+B$

 mostrar resultado

 volver al principio

Subrutina 1

$A = A - 10$

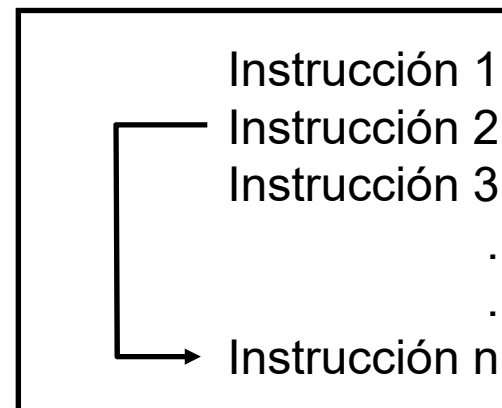
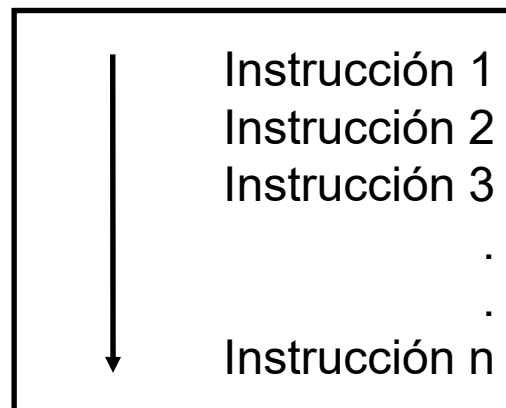
$B = B - 10$

Volver a principio programa

Nota: El pseudo código puede ser tan detallado como se quiera, hay muchos niveles de abstracción

ESTRUCTURA DE UN PROGRAMA

- Los pasos del algoritmo se expresan en los programas como instrucciones o sentencias.
- Por tanto, la ejecución de un programa consiste en la ejecución de forma secuencial de las instrucciones que lo componen.
- Esta secuencialidad puede verse rota por determinadas instrucciones.



PARTES PRINCIPALES DE UN PROGRAMA

Considerando como criterio de clasificación la función que realizan se pueden establecer 3 grupos de instrucciones:

- **Entrada de datos:** son las instrucciones que permiten transferir información de los dispositivos externos (teclado, ratón, etc.) hasta la memoria interna.
- **Proceso de datos:** indican cuales son las operaciones a realizar sobre los datos de entrada o resultados intermedios.
- **Salida de resultados:** toman los resultados de la memoria interna para representarlos en algún dispositivo externo de salida (monitor, impresora, etc.).

Estas instrucciones pueden aparecen entremezcladas en un programa.

CLASIFICACIÓN DE LAS INSTRUCCIONES

- No todos los lenguajes de programación disponen del mismo conjunto de instrucciones.
- La elaboración de un programa requiere conocer el repertorio de instrucciones del lenguaje y la forma de usarlas.
- En general, se pueden distinguir los siguientes tipos de instrucciones dentro de un programa:
 - Instrucciones de declaración.
 - Instrucciones de asignación.
 - Instrucciones de entrada.
 - Instrucciones de salida.
 - Instrucciones de control.
 - Instrucciones repetitivas.

COMPONENTES BÁSICOS DE UN PROGRAMA

- **Instrucciones:** Distintos comandos y operaciones que se pueden realizar.
- **Puntos de bifurcación:** permite la especificación de caminos alternativos:
si algo ocurre entonces camino 1
caso contrario camino 2
- **Bucles:** Parte de un programa que se repite un número de veces
 - **Bucles anidados:** bucles dentro de otros
 - **Contadores:** variables en un bucle cuyo valor se incrementa o decrementa una cantidad fija en cada iteración
 - **Acumulador:** Variable cuya función es almacenar el resultado de sumas sucesivas.

Ejemplo: Imprimir de 1 a 1000

BIBLIOGRAFÍA

- Joyanes, L., *Fundamentos de Programación, Algoritmos y Estructuras de Datos*. McGraw-Hill, 1996.
- Prieto, A., Lloris, A., Torres, J. C., *Introducción a la Informática*. 4ª Edición, McGraw-Hill, 2006.
- Galindo, J., Sánchez, P. J., Yáñez, A., del Jesús, M. J., Aguilera, J. J., Rodríguez, J. M., Sánchez, A., Argudo, J. F., *Fundamentos Informáticos*. 2ª Edición, Servicio de Publicaciones de la Universidad de Cádiz, 1997.