



Universidad Metropolitana
para la Educación y el Trabajo

Resolución de problemas
Lic. Diego Krauthamer

Agenda

- Resolución de problemas
- Algoritmos
- Técnicas de diagramación Lógica/Herramientas de Programación
- Ejemplo de aplicación ¿Cómo diseñar un algoritmo que sume dos números?
- Conclusiones
- Preguntas de revisión conceptual
- Bibliografía

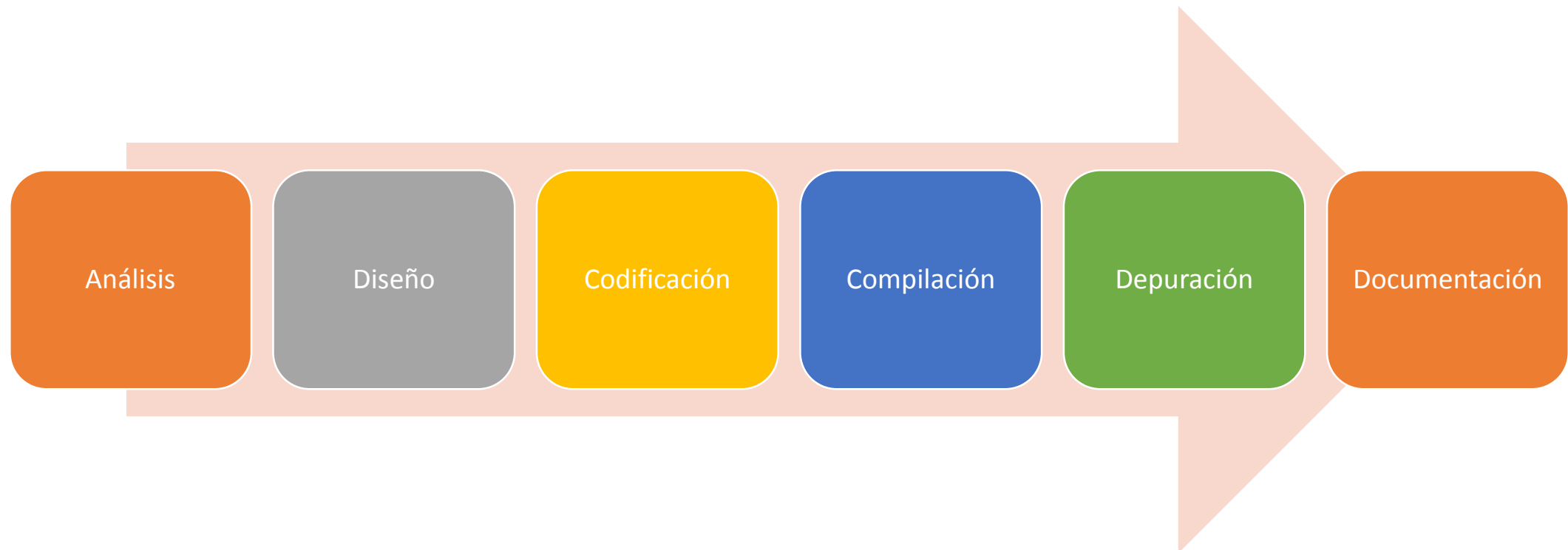
Resolución de problemas/1

“La resolución de problemas en un dispositivo consiste en la escritura de un programa y su ejecución”.

¹Programación en Turbo/Borland Pascal 7.0. Joyanes Aguilar. Editorial McGraw Hill. 1997. Cap1. Pág. 14



Resolución de Problemas/2 – Etapas



Algoritmo/1 - Concepto

“Un **algoritmo** es un método para resolver un problema mediante una serie de pasos precisos, definidos y finitos”.

¹Programación en Turbo/Borland Pascal 7.0. Joyanes Aguilar. Editorial Mc Graw Hill. 1997. Cap1. Pág. 15



Algoritmo/2 - Características

- ✓ **Preciso:** debe indicarse el orden de realización de cada paso.
- ✓ **Definido:** se debe obtener el mismo resultado cada vez que se ejecute.
- ✓ **Finito:** tiene un fin (número determinado de pasos).

Resolución de Problemas/3 – Análisis

- Debe contemplar **que debe hacer el programa y la solución deseada**.
- Dado que nos encontramos trabajando en un dispositivo electrónico es necesario definir “especificaciones detalladas de **entrada y salida**”.
- Para definir las especificaciones detalladas debemos preguntarnos:
 - ¿Qué entradas se requieren?
 - ¿Cuál/es es/son la/s salida/s deseada/s?
 - ¿Cuáles son los procesos requeridos?



Resolución de Problemas/4 – Diseño

- En esta etapa se determina **que hace el algoritmo**, a diferencia de la etapa anterior que nos preguntábamos como realiza el algoritmo la tarea solicitada.
- Se divide el problema a resolver en problemas más pequeños (Diseño Descendente o Top Down)



Resolución de Problemas/5 – Codificación

- En esta etapa “implementamos” el algoritmo en un lenguaje de programación determinado.
- Se divide el problema a resolver en problemas más pequeños (Diseño Descendente o Top Down)
- El diseño del algoritmo es independiente del lenguaje de programación en el que vaya a codificar posteriormente.



Resolución de Problemas/6 – Compilación/verificación

- La **compilación/verificación** consiste en la ejecución del programa con una variedad de datos entrada, denominados datos de prueba o test que determinaran si el programa tiene errores (bugs).
- La **depuración** es el proceso de encontrar errores y corregirlos o eliminarlos. Pueden encontrarse tres tipos de errores:
 - **De compilación:** suelen ser errores de sintaxis.
 - **De ejecución:** se producen por instrucciones que el dispositivo puede interpretar pero no ejecutar.
 - **Lógicos:** Se dan en la lógica del programa. Son los más difíciles de detectar.



Resolución de Problemas/7 – Documentación y mantenimiento

- La **documentación** de un problema es la descripción de los pasos a seguir en la resolución de un problema.
- La **documentación es vital** en cualquier software ya sea para realizar su mantenimiento, para corregir errores, o bien para incorporar nuevas funcionalidades al software.
- La documentación **se clasifica en**:
 - **Interna**: es la auto documentación código fuente (comentarios en las líneas de código).
 - **Externa**: incluye documentos de análisis, diagramas de flujo, pseudocódigo, manuales, etc.



Técnicas de Diagramación Lógica/1

- ✓ Las técnicas de diagramación lógica son herramientas que nos permiten especificar detalladamente el algoritmo.
- ✓ Se utilizan en la **etapa de diseño** de resolución de problemas.
- ✓ Algunas de las herramientas más utilizadas son el Diagrama de Flujo (Flowchart), el método Jackson y el **pseudocódigo**.



Técnicas de Diagramación Lógica/2 – Diagrama de Flujo

- ✓ Llamamos **diagrama de flujo** aquella representación gráfica utilizada para definir el algoritmo de un problema a fin de representar las distintas operaciones, datos o flujos mediante símbolos.
- ✓ Utiliza **símbolos estándar** definidos por el A.N.S.I (Instituto americano nacional de estándares)

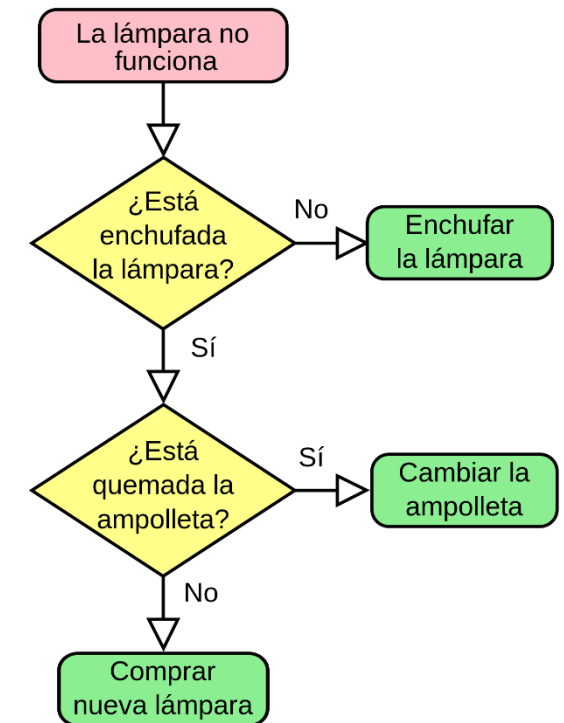


Figura – Diagrama de flujo

Técnicas de Diagramación Lógica/3 – Método Jackson

- ✓ Se trata de un **método de representación de programa en forma de árbol** denominado diagrama arborescente de Jackson.
- ✓ Un diagrama Jackson esta compuesto por:
 - ✓ Definición detallada de los datos de entrada y salida incluyendo los archivos lógicos utilizados.
 - ✓ Representación del proceso o algoritmo.

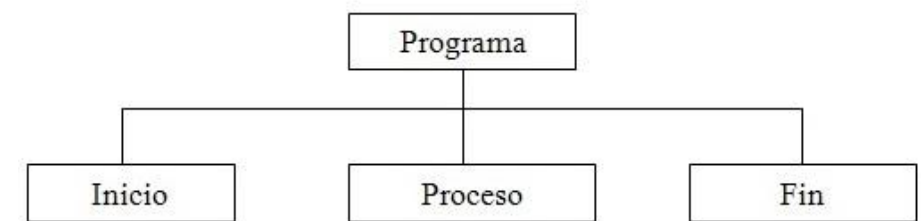


Figura – Método Jackson

Técnicas de Diagramación Lógica/4 – Pseudocódigo

“Es una herramienta de programación en la que las instrucciones se escriben en palabras similares al inglés o al español, que facilitan tanto la lectura como escritura de programas.

¹“Programación en Turbo/Borland Pascal 7.0. Joyanes Aguilar. Editorial Mc Graw Hill. 1997.

```
# Programa Ejemplo

#Declaración de Variables
numero=0

#Cuerpo del programa
Mostrar "Ingrese un número"
Leer numero

Mostrar numero
```

Figura – Pseudocódigo

Ejemplo de aplicación/1 - ¿Cómo diseñar un algoritmo que solicite un número entero y lo muestre por pantalla?

Pasos a seguir para Diseñar el Algoritmo:

1. ¿Cuáles son los datos de entrada que necesita el algoritmo?
2. ¿Cual/es es/son las salida/s que debo mostrar al usuario?
3. A partir del punto anterior ¿Cuáles procesos necesarios para obtener la/s salida/s del punto anterior?
4. Una vez que conozco los puntos 1,2, y 3 estoy listo/a para diseñar el algoritmo en alguna de las herramientas de diagramación lógica.

Ejemplo de aplicación/2 - ¿Cómo diseñar un algoritmo que solicite un número entero y lo muestre por pantalla? – Diseño del algoritmo

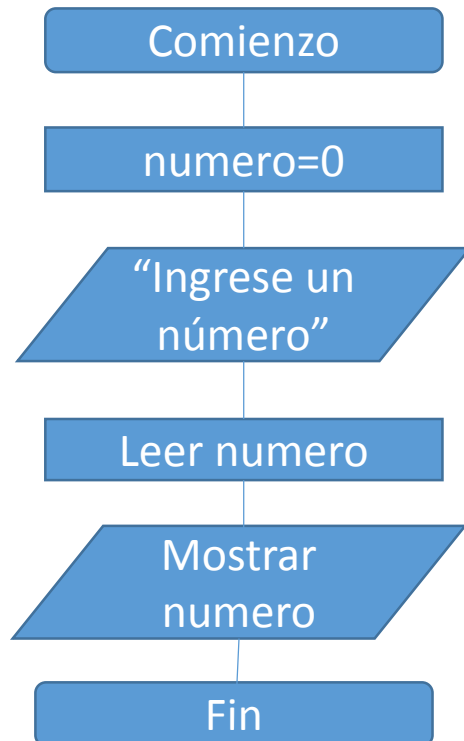


Figura – Diagrama de Flujo

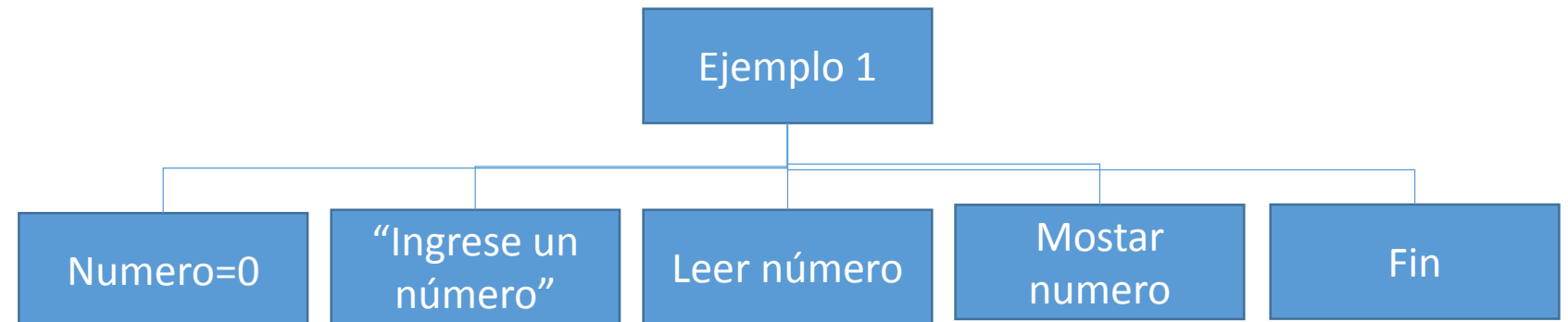


Figura – Método Jackson

Ejemplo de aplicación/3 - ¿Cómo diseñar un algoritmo que solicite un número entero y lo muestre por pantalla? – Diseño del algoritmo

```
# Programa Ejemplo

#Declaración de Variables
numero=0

#Cuerpo del programa
Mostrar "Ingrese un número"
Leer numero

Mostrar numero
```

Figura – Pseudocódigo del algoritmo

Conclusiones

- ✓ La resolución de problemas en un dispositivo (de escritorio, móvil, etc.) se realiza a través de algoritmos
- ✓ Una algoritmo es un proceso lógico que consta de un serie de pasos que se ejecutan en forma secuencial.
- ✓ El diseño de un algoritmo se realiza mediante una técnica de diagramación.
- ✓ El uso de una técnica de diagramación lógica me permite “independizarme” del lenguaje de programación. Además constituye una “buena práctica” (Best practices) en programación.
- ✓ La implementación del algoritmo se realiza a partir de la codificación del algoritmo partiendo de la técnica de diagramación lógica empleada.

Preguntas de revisión conceptual.

1. Enumere y explique brevemente las etapas del proceso de resolución de problemas.
2. ¿Qué es un algoritmo?
3. ¿Qué es la diagramación Lógica?
4. ¿Qué es el diagrama de flujo?
5. ¿Qué es el método Jackson?
6. ¿Qué es el pseudocódigo?
7. Realice la diagramación lógica de un programa que calcule la suma de dos números, y que la muestre por pantalla. Utilice diagrama de flujo, método Jackson y pseudocódigo.

Bibliografía obligatoria

- Programación en Turbo/Borland Pascal 7.0. Joyanes Aguilar. Editorial Mc Graw Hill. 1997.
Capítulo 1.