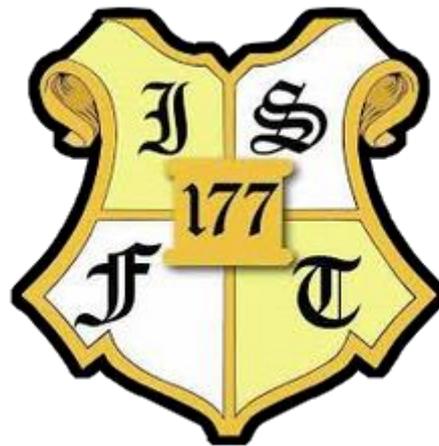


# **INSTITUTO SUPERIOR DE FORMACIÓN TÉCNICA N°177**



## **Algoritmos y Estructuras de Datos 1**

### **Conceptos básicos**

**Prof. Lic. Walter Carnero**  
profewaltercarnero@gmail.com



---

## APUNTE: AyEDD01 – CONCEPTOS BÁSICOS

---

### INTRODUCCIÓN

En el espacio de formación algoritmos y estructuras de datos desarrollaremos habilidades de resolución de problemas, orientando dichas habilidades al pensamiento lógico para buscar la mejor solución al problema que se presente. El principal objetivo de la cátedra es que las habilidades y destrezas obtenidas puedan ser aplicadas a la programación de sistemas.

Un problema se puede definir como una situación en la cual se trata de alcanzar una meta y para lograrlo se deben hallar y utilizar determinados medios y estrategias. La mayoría de los problemas tienen algunos elementos en común: un estado inicial; un final, lo que se pretende lograr; un conjunto de recursos, lo que está permitido hacer y/o utilizar; y un estado actual de conocimientos con todo ello podremos, aplicando cierta metodología, resolver el problema planteado.

Cada disciplina dispone de estrategias específicas para resolver problemas de su ámbito; por ejemplo, resolver problemas matemáticos implica utilizar estrategias propias de las matemáticas. Sin embargo, algunos psicólogos opinan que es posible utilizar con éxito estrategias generales, útiles para resolver problemas en muchas áreas, algunos de esos métodos o estrategias generales son:

- Algoritmos.
- Modelo de procesamiento de información.
- Análisis de medios y fines.
- Lluvia de ideas.
- Procesamiento lateral.

Nosotros nos centraremos en el primero, es decir utilizaremos estrategias algorítmicas para la resolución de problemas. Ahora bien, antes de empezar vamos a definir que es un algoritmo. El vocablo es de origen árabe (al-jwarizmi)<sup>1</sup> define en un conjunto finito de instrucciones los pasos a seguir para resolver un problema determinado. Si aplicamos un algoritmo a la resolución de un problema mediante el uso de una computadora estaremos creando un programa.

---

<sup>1</sup> Abu Abdallah Muhammad ibn Mūsā al-Jwārizmī, conocido generalmente como al-Juarismi, fue un matemático, astrónomo y geógrafo persa musulmán, que vivió aproximadamente entre 780 y 850.



Un programa justamente es un conjunto ordenado de acciones que le indican a la computadora cuales son los pasos que se deben seguir para realizar la concreción de una determinada tarea. Un programa define los pasos a seguir mediante la aplicación de algún algoritmo. Para poder diseñar nuestros programas es importante que comprendamos cuál es el problema al que debemos darle solución. Las estrategias de resolución de problemas generalmente presentan cuatro operaciones:

1. Entender el problema.
2. Diseñar y trazar un plan.
3. Ejecutar el plan, resolverlo.
4. Revisar la solución.

La aplicación práctica para diseñar un algoritmo o un programa utilizando la estrategia mencionada sería:

### **1. Entender el problema**

- a. Leer la consigna varias veces.
- b. Establecer cuáles son los datos.
- c. Organizar la información.
- d. Agrupar los datos en categorías o estructuras.
- e. Trazar un diagrama de flujo de datos.

### **2. Diseñar y trazar un plan**

- a. Definir y escoger las operaciones a realizar.
- b. Descompones, si es posible, en problemas más pequeños (modularizar).
- c. Eliminar datos inútiles.

### **3. Ejecutar el plan, resolverlo**

- a. Ejecutar paso a paso la solución.
- b. Determinar qué se puede simplificar.
- c. Trazar diagrama de flujo de datos.

### **4. Revisar la solución**

- a. Probar la solución y verificar los resultados.
- b. Realizar las correcciones necesarias.

A modo de resume y aclarando algunos conceptos mencionados diremos que:



**Algoritmo:** define en un conjunto finito de instrucciones los pasos a seguir para resolver un problema.

**Programa:** Es un conjunto ordenado de acciones que le indican a la computadora cuales son los pasos que se deben seguir para resolver un problema, un programa se escribe mediante un lenguaje de programación utilizando un algoritmo.

Se puede decir que un programa almacena la inteligencia humana ya que en la resolución de un problema se aplican las habilidades y destrezas que son propias de cada programador.

### Concepto de variable

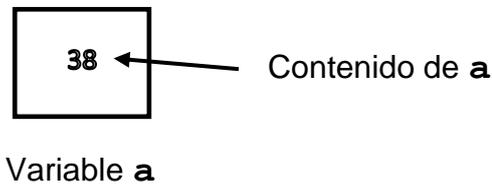
El diseño de algoritmos y la programación se basan en la utilización de variables, una variable es un elemento al cual le damos un nombre y le atribuimos un determinado tipo de información, más técnicamente, En programación cuando declaramos una variable reservamos un espacio en la memoria de la computadora para almacenar un dato del tipo con el cual fue declarado. En todos los lenguajes de programación existen distintos tipos de datos, de variables, es decir que en algunas podremos almacenar números, en otras letras y en otras hasta una cadena de caracteres o párrafos completos. Un ejemplo de variables sería:

$$a \leftarrow 38$$

En el ejemplo tenemos una variable con el nombre **a** y a la misma le asignamos el 38, esto quiere decir que la variable **a** es de tipo numérica. Siempre la variable declarada va del lado izquierdo y el contenido o lo que “asignamos” a la variable va del lado derecho, en el ejemplo la asignación se realizó con una flecha ( $\leftarrow$ ), cuando escribimos un programa generalmente la asignación se realiza con el signo igual (=) es decir que en un programa sería:

$$a = 38;$$

Para aclarar un poco más lo expuesto, y recordando que al declarar una variable reservamos espacio en la memoria de programa de la computadora, podemos establecer la analogía con una caja en la cual guardamos contenido, por ejemplo, aplicando esta idea al caso de la variable **a** podemos expresar:



Una variable puede recibir cualquier nombre siguiendo algunas reglas simples, puede contener números en el nombre, pero no puede comenzar con un número, no puede tener letras ñ o Ñ y algunos caracteres como / o &. A continuación, algunos ejemplos:

Nombres de variables válidos	Nombres de variables no validos
<code>un_numero</code>	<code>año</code> (no es válida porque tiene ñ)
<code>num1</code>	<code>número</code> (no es válida porque tiene acento)
<code>Nombre</code>	<code>1numero</code> (no es válida porque comienza con un número)
<code>apellido1</code>	<code>Un numero</code> (no es válida porque tiene un espacio)
<code>apellido_2</code>	
<code>numero</code>	
<code>Pi</code>	
<code>Temp</code>	
<code>anio</code>	

Además, cuando programamos cada lenguaje de programación tiene palabras reservadas, es decir palabras que no pueden utilizarse ya que el propio lenguaje las utiliza en su sintaxis.

Se mencionó anteriormente que una variable puede guardas cierto tipo de información, ese tipo de información está dada por:

- Números enteros.
- Números en punto flotante (con decimales).
- Booleanos (verdadero o falso).



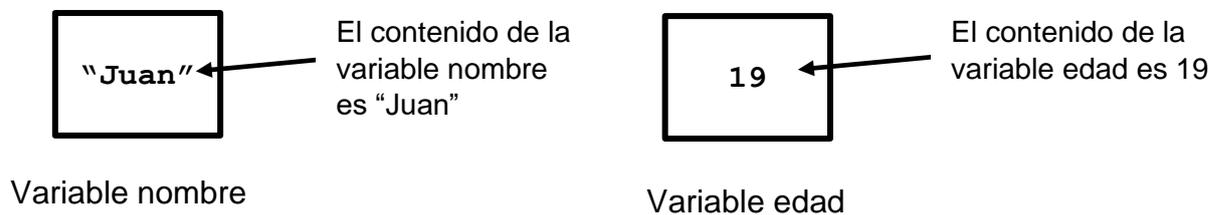
- Cadenas de caracteres (string)
- Caracteres individuales.

En base a lo expuesto no es lo mismo una variable donde guardaremos por ejemplo el nombre de una persona que aquella donde guardaremos su edad, un ejemplo podría ser:

nombre = "Juan";

edad = 19;

Utilizando la analogía de los cajoncitos:



Observen que las variables a las cuales se les asignan cadenas de caracteres (palabras) esas cadenas deben estar entre comillas. Y para el caso que sea una sola letra, un solo símbolo se indicará entre comillas simples, por ejemplo:

```
letra = 'E' ;
```

Se declara una variable con el nombre **letra** y se le asigna como contenido la letra **'E'** .

## BIBLIOGRAFÍA

Deitel, H. & Deitel, P. (2005). *"Cómo programar en C/C++ y Java"*. Pearson Educación. México.

Savitch, W. (2007). *"Resolución de problemas con C++"*. Pearson Educación. México.