

# Unidad I

## Fundamentos de estructura de datos

### 1.1. Definición.

En programación, una estructura de datos es una forma de organizar un conjunto de datos elementales con el objetivo de facilitar su manipulación. Un dato elemental es la mínima información que se tiene en un sistema.

Una estructura de datos define la organización e interrelación de estos y un conjunto de operaciones que se pueden realizar sobre ellos. Las operaciones básicas son:

- Alta, adicionar un nuevo valor a la estructura.
- Baja, borrar un valor de la estructura.
- Búsqueda, encontrar un determinado valor en la estructura para realizar una operación con este valor, en forma secuencial o binario (siempre y cuando los datos estén ordenados).

Otras operaciones que se pueden realizar son:

- Ordenamiento, de los elementos pertenecientes a la estructura.
- Apareo, dadas dos estructuras originar una nueva ordenada y que contenga a las apareadas.

Cada estructura ofrece ventajas y desventajas en relación a la simplicidad y eficiencia para la realización de cada operación. De esta forma, la elección de la estructura de datos apropiada para cada problema depende de factores como la frecuencia y el orden en que se realiza cada operación sobre los datos.

### 1.2. Clasificación.

Una estructura de datos es una clase de datos que se puede caracterizar por su organización y operaciones definidas sobre ella. Algunas veces a estas estructuras se les llama tipos de datos.

en ellas encontramos las siguientes:

#### ESTRUCTURAS LÓGICAS DE DATOS:

En un programa, cada variable pertenece a alguna estructura de datos explícita o implícitamente definida, la cual determina el conjunto de operaciones validas para

ella. Las estructuras de datos que se discuten aquí son estructuras de datos lógicas. Cada estructura de datos lógica puede tener varias representaciones físicas diferentes para sus almacenamientos

### **1.3. Estructuras lineales y no lineales.**

Las estructuras de datos simples se pueden combinar de varias maneras para formar estructuras más complejas. Los dos casos principales de estructuras de datos son las lineales y las no lineales, dependiendo de la complejidad de las relaciones lógicas que representan. Las estructuras de datos lineales incluyen pilas, colas y listas ligadas lineales. Las estructuras de datos no lineales incluyen grafos y árboles.

Datos estáticos: su tamaño y forma es constante durante la ejecución de un programa y por tanto se determinan en tiempo de compilación. El ejemplo típico son los arrays. Tienen el problema de que hay que dimensionar la estructura de antemano, lo que puede conllevar desperdicio o falta de memoria.

Datos dinámicos: su tamaño y forma es variable (o puede serlo) a lo largo de un programa, por lo que se crean y destruyen en tiempo de ejecución. Esto permite dimensionar la estructura de datos de una forma precisa: se va asignando memoria en tiempo de ejecución según se va necesitando.

### **1.4. Estructuras dinámicas y estáticas.**

Son aquellas en las que el tamaño ocupado en memoria se define antes de que el programa se ejecute y no puede modificarse dicho tamaño durante la ejecución del programa.

Estas estructuras están implementadas en casi todos los lenguajes.

Su principal característica es que ocupan solo una casilla de memoria, por lo tanto una variable simple hace referencia a un único valor a la vez, dentro de este grupo de datos se encuentra:

- a) Enteros
- b) Reales
- c) Caracteres
- d) Booleanos
- e) Enumerados
- f) Subrangos

Nota: Los últimos no existen en algunos lenguajes de programación.

Estructura de Datos Dinamicas:

No tienen las limitaciones o restricciones en el tamaño de memoria ocupada que son propias de las estructuras estáticas.

Mediante el uso de un tipo de datos específico, denominado puntero, es posible construir estructuras de datos dinámicas que no son soportadas por la mayoría de los lenguajes, pero que en aquellos que si tienen estas características ofrecen soluciones eficaces y efectivas en la solución de problemas complejos.

Se caracteriza por el hecho de que con un nombre se hace referencia a un grupo de casillas de memoria.

Es decir un dato estructurado tiene varios componentes.