

Principios del diseño de Estructura de datos

Capítulo 1

Abstracción de datos

- ▶ Para realizar un programa complicado, debemos dividirlo en partes que resuelvan sub problemas del problema original.
- ▶ En programación se usa la abstracción como una forma de descomponer un problema.
- ▶ El método de descomposición en base a abstracciones se continua aplicando hasta llegar a operaciones lo bastante sencillas como para ser programadas de forma directa.
- ▶ Para la resolución y optimización

Definición

- ▶ Una estructura de datos es una colección de valores, la relación que existe entre los valores, y las operaciones que podemos hacer sobre ellos.
- ▶ Una estructura de datos describe el formato en que los valores van a ser almacenados, cómo van a ser accedidos y modificados, pudiendo así existir una gran cantidad de estructuras de datos.
- ▶ Ejemplo: Registro de estudiantes en una materia.

- ▶ Las estructuras de datos se estudian como conceptos sobre programación, y no sobre un lenguaje de programación en específico.
- ▶ Por lo que cada lenguaje puede tener diferentes implementaciones de estructuras de datos.
- ▶ Las estructuras de datos son útiles porque siempre manipularemos datos, y si los datos están organizados esta tarea será mucho más fácil.

Tipos de datos

- ▶ Un tipo de datos determina un conjunto de valores ya sean constantes o variables.
- ▶ Existen tipos de datos estándar que son definidos por muchos lenguajes de programación, estos tipos de datos son los siguientes.
 - ▶ Integer. Comprende un conjunto de números enteros; dependiendo de la potencia del compilador o computador, el rango de números enteros que abarca puede variar.
 - ▶ Real. Es un subconjunto de los reales, pero puede ser inexacta por el redondeo en las operaciones de cálculo.
 - ▶ Boolean. Solo tiene dos valores que son TRUE y FALSE.
 - ▶ Char. Comprende los valores imprimibles.
 - ▶ String. Comprende un conjunto o agrupación de caracteres

Tipos de estructuras de datos

- ▶ Al hablar de estructuras de datos debemos pensar en primera instancia en como los datos se representan en la memoria, al partir de esta pregunta podemos darnos la idea correcta sobre la base de nuestra estructura y cómo los datos se van a almacenar.
- ▶ **Lineales.** Cada nodo tiene un sucesor, por ejemplo una lista de estudiantes (se pueden implementar con arrays, matrices, listas enlazadas, etc).
- ▶ **Ramificadas.** Por ejemplo un árbol genealógico en el cual se puedan establecer, abuelos, padres, hijos, nietos, etc.

Estructuras de datos que se estudiarán

- ▶ Pilas
- ▶ Colas
- ▶ Listas
- ▶ Multilistas
- ▶ Grafos
- ▶ Árboles binarios
- ▶ Árboles generales

Operaciones que se realizan con estas estructuras de datos

- ▶ Insertar
- ▶ Eliminar
- ▶ Vacía
- ▶ Llena
- ▶ Buscar un elemento
- ▶ Ordenar
- ▶ Otras (según el la estructura de datos que se vaya a implementar)

Lenguaje de programación

- ▶ El lenguaje de programación para la implementación de Estructuras de datos será C#, utilizando el entorno de desarrollo visual de Visual Studio .Net