

Tipos De Datos Abstractos

Fernel Moreno

2024-03-03

¿Que es un tipo de dato abstracto o TDA?

Un TDA, o Tipo de Dato Abstracto, es un modelo matemático para tipos de datos, donde un tipo de datos se define por su comportamiento (semántica) desde el punto de vista de un usuario de datos, en términos de posibles valores, posibles operaciones sobre estos datos y el comportamiento de estas operaciones. Esta definición es independiente de cómo se implementan los datos y las operaciones en una computadora. Los TDA se utilizan para especificar las interfaces de los datos y sus operaciones sin tener en cuenta los detalles de su implementación, permitiendo de esta manera que los detalles de implementación puedan cambiarse sin modificar la forma en que los datos son vistos desde fuera. Esto proporciona una abstracción de los datos que ayuda a manejar la complejidad en los sistemas de software.

¿Que Funcionalidades Brinda?

- **Encapsulamiento:** Los TDA ocultan los detalles de la implementación del usuario, exponiendo solo las operaciones relevantes para el uso de los datos. Esto asegura que los datos no puedan ser manipulados de formas no previstas, manteniendo la integridad de los mismos.
- **Reusabilidad:** Al definir operaciones de manera abstracta, los TDA permiten que se reutilicen en diferentes contextos. Una vez que un TDA está implementado, puede ser utilizado en cualquier programa sin necesidad de reescribir el código.
- **Mantenibilidad:** Los cambios en la implementación de un TDA (por ejemplo, para optimización) no afectan al código que utiliza el TDA, siempre y cuando la interfaz se mantenga. Esto facilita la actualización y mantenimiento del código.
- **Abstracción:** Permiten al desarrollador pensar en un nivel más alto de abstracción. En lugar de preocuparse por detalles de implementación, el desarrollador puede concentrarse en cómo usar estos tipos de datos para resolver problemas complejos.

¿Cómo se aplican en el desarrollo de un programa?

Diseño: En la fase de diseño de un programa, se identifican los tipos de datos necesarios y se definen sus operaciones sin preocuparse por los detalles de su implementación. Esto ayuda a esclarecer qué debe hacer el programa antes de cómo lo hará.

Implementación: Los TDA se implementan utilizando estructuras de datos concretas y algoritmos que realizan las operaciones definidas. Por ejemplo, un TDA "Lista" puede implementarse como una lista enlazada o un arreglo dinámico, pero los detalles de cómo se añaden o eliminan elementos de la lista son ocultos al usuario del TDA.

Uso en el Programa: En el código del programa, los desarrolladores utilizan los TDA a través de sus interfaces. Esto significa que trabajan con operaciones como **añadir**, **eliminar**, **buscar**, etc., sin necesidad de conocer cómo estas operaciones se llevan a cabo internamente.

Mantenimiento y Evolución: Si se necesita cambiar la implementación de un TDA (por ejemplo, para mejorar el rendimiento), esto se puede hacer sin afectar el resto del programa, siempre que la interfaz del TDA se mantenga constante. Esto facilita la evolución y el mantenimiento del software a lo largo del tiempo.

Referencias

1. <https://sites.google.com/site/programacioniiuno/temario/unidad-2---tipo-abstracto-de-dato/tipo-de-dato-abstracto>
2. https://es.wikipedia.org/wiki/Tipo_de_dato_abstracto
3. <https://www.infor.uva.es/~mserrano/EDI/cap2.pdf>