**1.2 – Tipos de datos abstractos (TDA).**

**Qué son los Tipos de Datos Abstractos**

Un TAD se define como una estructura algebraica compuesta por un conjunto de objetos abstractos que modelan elementos del mundo real, y un conjunto de operaciones para su manipulación.

Un TAD es un ente cerrado y autosuficiente, que no requiere de un contexto específico para que pueda ser utilizado en un programa. Esto garantiza portabilidad y reutilización del software. Las partes que forman un TAD son:

1. Atributos (tipos de datos, identificadores, etc.)
2. Funciones (rutinas) que definen las operaciones válidas para manipular los datos (atributos).

Las operaciones de un TAD se clasifican en 3 grupos, según su función sobre el objeto abstracto:

* Constructora: es la operación encargada de crear elementos del TAD.
* Modificadora: es la operación que puede alterar el estado de un elemento de un TAD.
* Analizadora: es una operación que no altera el estado del objeto, sino que tiene como misión consultar su estado y retornar algún tipo de información.

**Cuáles son los TDA que maneja C#**

Entre las aplicaciones de los TAD’s se encuentran el TAD lista, el TAD pila, el TAD cola, etc.

**TAD’s Estáticos.**

La creación y mantenimiento de un TAD estático requiere de memoria no dinámica, es decir, el espacio en memoria para almacenar los datos es reservado en tiempo de compilación (Arreglos).

**TAD’s Dinámicos (Asignación dinámica de memoria).**

La creación y mantenimiento de estructuras dinámicas de datos (TAD’s dinámicos), requiere de obtener más espacio de memoria (reservar memoria) en tiempo de ejecución para almacenar datos o para almacenar el tipo de clase “Nodo”.

**El TAD Lista**

Una lista está formada por una serie de elementos llamados nodos los cuales son objetos que contiene como variable miembro un puntero asignado y variables de cualquier tipo para manejar datos.

El puntero sirve para enlazar cada nodo con el resto de nodos que conforman la lista. De esto podemos deducir que una lista enlazada (lista) es una secuencia de nodos en el que cada nodo esta enlazado o conectado con el siguiente (por medio del puntero mencionado anteriormente). El primer nodo de la lista se denomina cabeza de la lista y el último nodo cola de la lista. Este último nodo suele tener su puntero igualado a NULL Para indicar que es el fin de la lista.

**El TAD pila.**

Una pila (stack) es una secuencia de cero o más elementos de un mismo tipo, que puede crecer y decrecer por uno de sus extremos (el tope de la pila).

Las pilas se denominan también estructuras LIFO (Last In First Out), porque su característica principal es que el último elemento en llegar es el primero en salir. Son muy utilizadas en programación para evaluar expresiones.

**Programa ejemplo en C# que utilice un TDA.**

Implementación de pilas a través de TAD predeterminado por C#.

Cree un proyecto modo consola en Visual C# y agregue al inicio el nombre de espacio Collections de la siguiente manera: Using System.Collections;

string valor;

Stack mipila = new Stack( );

//Ingreso de elementos a la pila

mipila.Push(“z”);

mipila.Push(“ba”);

mipila.Push(“nom”);

//imprimir elementos de la pila pueden ser de tipo char, int, string o

//cualquier otro tipo ya que es un tipo object dependerá de lo que el

//usuario necesite

foreach (string var in mipila)

{

Console.WriteLine(var);

}

Console.WriteLine(“\n\n”);

//Peek Retorna el valor que está al tope de la pila sin eliminarlo

Console.WriteLine(“El tope de la pila es”);

Console.WriteLine(“mipila.Peek()”);

//retorna el valor del tope eliminándolo

valor = mipila.Pop().ToString();

Console.WriteLine(“eliminado de la pila” + valor);

Console.WriteLine(“\n\n”);

//mostrando contenido de la pila

foreach (string var in mipila)

{

Console.WriteLine(var);

}

Console.ReadLine();

<http://www.cartagena99.com/recursos/programacion/apuntes/Tema1TADs.pdf>

<https://docplayer.es/44195023-Tema-tipos-abstractos-de-datos-tad-s-en-c.html>

<https://users.dcc.uchile.cl/~bebustos/apuntes/cc30a/TDA/#:~:text=Un%20Tipo%20de%20dato%20abstracto,como%20est%C3%A9n%20implementadas%20dichas%20operaciones>.