



Instituto Nacional de Formación
Técnico Profesional
Infotep.gob.do

Tecnico Ciberseguridad
Profesor: Kevin Feliz Henriquez
Nombre: Joice Pérez Paulino



Dapiato Prosalud
A la vanguardia de la tecnologia
secomdr.com

Estructura de datos actividad 1E

Ciberseguridad

Estructura de datos

Actividad #1

Las Estructuras de datos son formas de organizar y almacenar información para que pueda ser utilizada de manera eficiente.

Cada estructura tiene sus propias ventajas y desventajas, lo que las hace más adecuadas para ciertas tareas que para otras.

Listas y arreglos Es una colección ordenada de elementos, donde cada elemento tiene una posición (o índice) que comienza desde 0. Son de las estructuras más simples y usadas. Su tamaño puede ser fijo o dinámico.

Concepto un conjunto de cosas una al lado de la otra, cada una con un número de posición. Ejemplo en JavaScript

JavaScript

```
let frutas = ["manzanas", "banana", "cerezas"];  
console.log(frutas[0]); // salida: manzana
```

Listas Enlazadas Es una colección de elementos (nodos) donde cada nodo contiene el dato y un enlace al siguiente nodo de la secuencia. A diferencia de un arreglo, los elementos no están almacenados en posiciones de memoria contiguas.

Concepto Una cadena de personas, donde cada persona solo sabe quién es la siguiente.

- * **Ventajas** Permite inserciones y eliminaciones rápidas en cualquier punto.
- * **Desventaja** El acceso a un elemento específico es más lento, ya que hay que recorrer la lista desde el principio.

Diccionarios

Un **Diccionario** (o **mapa**) es una estructura que almacena datos en pares de **Clave-Valor**. Cada clave es única y se utiliza para acceder a su valor asociado.

Concepto Una guía telefónica, donde la clave es el nombre y el valor es el número de teléfono.

Ejemplo

```
let persona = {  
  nombre: "Carlos",  
  edad: 30,  
  ciudad: "Bogotá"  
};  
console.log(persona.nombre); // salida: "Carlos"
```

Tablas Hash Hashes

Una **tabla Hash** es una estructura de datos que implementa un diccionario, utilizando una función hash para calcular un índice a partir de una clave. Esto permite un acceso muy rápido a los datos.

Concepto: Un casillero donde la llave de tu código (clave) te dice en qué casillero (índice) se encuentra tu objeto (valor). **Ventajas:** Búsqueda, inserción y eliminación de elementos en tiempo constante $O(1)$.

Pilas

Una **pila** es una estructura de datos que sigue el principio **LIFO** (last in, first out), es decir, el último elemento que entra es el primero en salir.

Concepto Una pila de platos: El último plato que pones encima es el primero que tomas.

Operaciones clave:

- * **Push:** Añadir un elemento.
- * **Pop:** Eliminar el último elemento añadido

Ejemplo:

```
let pila = [];  
pila.push("libro 1"); // Añadir  
pila.push("libro 2");  
console.log(pila.pop()); // salida: "libro 2"
```


Estructura de datos actividad 1E

Colas

Una cola es una estructura de datos que sigue el principio FIFO (first in, first out), es decir, el primer elemento que entra es el primero en salir.

Concepto: La cola del supermercado, la primera persona en la fila es la primera en ser atendida. Operaciones Claves:

- * enqueue: Añadir un elemento al final de la cola.
- * dequeue: Eliminar el primer elemento de la cola.

Ejemplo (Colas)

```
let colaDeEspera = [];  
// Enqueue (agregar a la cola)  
colaDeEspera.push("Cliente A");  
colaDeEspera.push("Cliente B");  
console.log(colaDeEspera); // Salida: ["Cliente A", "Cliente B", "Cliente C"]  
// Dequeue (atender al cliente)  
let primerCliente = colaDeEspera.shift();  
console.log(primerCliente); // Salida: "Cliente A"  
console.log(colaDeEspera); // Salida: ["Cliente B", "Cliente C"]
```

Árbol Binario

Es una estructura de datos jerárquica, donde cada nodo tiene, como máximo, dos nodos hijos: uno a la izquierda y uno a la derecha.

Concepto: Un árbol familiar, donde cada persona (nodo) tiene dos hijos.

Uso: Se utiliza comúnmente para búsquedas eficientes y organización de datos jerárquicos.

Códigos Actividad 1

```
html  
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
<title>Document</title>  
<script type="module" src="app.js"></script>  
</head>  
<body>  
  <h1>Presiona la tecla F12. Luego haz click en la herramienta de consola para ver la ejecución del código JavaScript.</h1>  
</body>  
</html>
```

app.js

```
console.log("---- 1. Arreglos ----");  
const numeros = [10, 20, 30, 40, 50];  
numeros.push(60);  
console.log("Arreglos modificado: ", numeros);  
console.log("Elemento en índice 2: ", numeros[2]);
```

```
console.log("\n--- 2. Diccionario (objetos) ---");  
const libro = {  
  titulo: "El quijote",  
  autor: "Miguel de Cervantes",  
  año: 1605  
};
```

```
console.log("Título del libro: ", libro.titulo);  
libro.editorial = "Anaya";  
console.log(libro);  
console.log("\n--- 3. Pilas (stacks) ---");  
const pila = [];  
pila.push("dato1");  
pila.push("dato2");  
pila.push("dato3");  
console.log("Pila después de apilar: ", pila);  
const datoDesapilado = pila.pop();  
console.log("Dato desapilado: ", datoDesapilado);  
console.log("Pila después de desapilar: ", pila);  
console.log("\n--- 4. Colas (Queues) ---");  
const cola = [];  
cola.push("cliente1");  
cola.push("cliente2");  
cola.push("cliente3");  
console.log("Cola después de encolar: ", cola);  
const datoDeCola = cola.shift();  
console.log("Dato de cola: ", datoDeCola);  
console.log("Cola después de decolar: ", cola);
```

Ejecucion de los códigos de la actividad1E

Preciona la tecla F12, luego haz click en la herramienta de consola para ver la ejecución del código javascript.

