



Universidad Nacional Autónoma de México

Facultad de Estudios Superiores Aragón

Ingeniería en Computación

Asignatura: Estructura de datos

TAREA 9: Evaluación de balanceo de paréntesis y llaves

Profesor: Jesús Hernández Cabrera

Alumno: Juan Diego Ortiz Cruz

Grupo: 1360

Fecha: 24/09/2024

Capturas Clase pilaADT:

```
1  export class PilaADT<E> {
2      private datos: ListaLigada<E>;
3
4      constructor() {
5          this.datos = new ListaLigada<E>();
6      }
7
8      public estaVacia(): boolean {
9          return this.datos.getLongitud() === 0;
10     }
11
12     public longitud(): number {
13         return this.datos.getLongitud();
14     }
15
16     public peek(): E | null {
17         return this.datos.getPrimero();
18     }
19
20     public push(elemento: E): void {
21         this.datos.agregarAlInicio(elemento);
22     }
23
24     public pop(): E | null {
25         return this.datos.eliminarElPrimero();
26     }
27
28     public toString(): string {
29         if (this.estaVacia()) {
30             return "";
31         }
32
33         let result = "";
34         let current = this.datos.getPrimero();
35         let index = 0;
36
37         while (current !== null) {
38             result += current?.toString();
39             current = this.datos.obtener(index + 1);
40             index++;
41         }
42         return result;
43     }
44 }
```


Captura clase VerificadorBalanceo:

```
1  export class VerificadorBalanceo {
2      private static esApertura(caracter: string): boolean {
3          return caracter === '(' || caracter === '{';
4      }
5
6      private static esCierre(caracter: string): boolean {
7          return caracter === ')' || caracter === '}';
8      }
9
10     private static coinciden(apertura: string, cierre: string): boolean {
11         return (apertura === '(' && cierre === ')') || (apertura === '{' && cierre === '}');
12     }
13
14     private static obtenerEsperado(apertura: string): string {
15         let esperado = '';
16         switch (apertura) {
17             case '(': esperado = ')'; break;
18             case '{': esperado = '}'; break;
19         }
20         return esperado;
21     }
22
23     public static verificarBalanceo(texto: string): boolean {
24         const pila = new PilaADT<string>();
25
26         for (let i = 0; i < texto.length; i++) {
27             const caracter = texto[i];
28             if (this.esApertura(caracter)) {
29                 pila.push(caracter);
30                 console.log(`Caracter ${i}: '${caracter}' - Añadido a la pila. Pila actual: ${pila}`);
31             } else if (this.esCierre(caracter)) {
32                 if (pila.estaVacia()) {
33                     console.log(`Caracter ${i}: '${caracter}' - Error: No se esperaba un '${caracter}'`);
34                     return false;
35                 }
36                 const ultimaApertura = pila.pop();
37                 if (!this.coinciden(ultimaApertura!, caracter)) {
38                     console.log(`Caracter ${i}: '${caracter}' - Error: No coincide con la última apert`);
39                     return false;
40                 }
41                 console.log(`Caracter ${i}: '${caracter}' - Coincide con '${ultimaApertura}'. Removido`);
42             }
43         }
44
45         if (!pila.estaVacia()) {
46             const esperado = this.obtenerEsperado(pila.peek());
47             console.log(`Error: Se esperaba un '${esperado}'`);
48             return false;
49         }
50
51         return pila.estaVacia();
52     }
53 }
```

Capturas Clase Main:

```
1 import { VerificadorBalanceo } from "./VerificadorBalanceo.js";
2 import * as fs from 'fs/promises';
3 import * as path from 'path';
4
5 async function leerArchivo(rutaArchivo: string): Promise<string> {
6     try {
7         const contenido = await fs.readFile(rutaArchivo, 'utf-8');
8         return contenido;
9     } catch (error) {
10         console.error(`Error al leer el archivo: ${error}`);
11         return '';
12     }
13 }
14
15 async function main() {
16     const rutaArchivo = path.join(process.cwd(), 'programa_suma.c');
17
18     try {
19         const contenidoPrograma = await leerArchivo(rutaArchivo);
20
21         if (contenidoPrograma) {
22             console.log(`Archivo: "${rutaArchivo}"`);
23             const balanceado = VerificadorBalanceo.verificarBalanceo(contenidoPrograma);
24             console.log(`Resultado final: ${balanceado ? 'Balanceado' : 'No balanceado'}`);
25         } else {
26             console.log("No se pudo leer el contenido del archivo.");
27         }
28     } catch (error) {
29         console.error(`Ocurrió un error: ${error}`);
30     }
31 }
32
33 main();
```

Código de prueba en C:



```
1  #include <stdio.h>
2
3  // Función para sumar dos números enteros
4  int suma(int a, int b) {
5      return a + b;
6  }
7
8  // Función principal
9  int main() {
10     int num1 = 5;
11     int num2 = 3;
12     int resultado;
13
14     // Llamada a la función suma
15     resultado = suma(num1, num2);
16
17     // Imprimir el resultado
18     printf("La suma de %d y %d es: %d\n", num1, num2, resultado);
19
20     // Probar la función con otros valores
21     printf("La suma de %d y %d es: %d\n", 10, 20, suma(10, 20));
22
23     return 0;
24 }
```

Captura ejecucion(código correcto en C) consola:

```
Archivo: "C:\Users\juani\OneDrive\Escritorio\Estructura-de-datos\tarea9\programa_suma.c"
Character 73: '(' - Añadido a la pila. Pila actual: (
Character 86: ')' - Coincide con '('. Removido de la pila. Pila actual:
Character 88: '{' - Añadido a la pila. Pila actual: {
Character 110: '}' - Coincide con '{'. Removido de la pila. Pila actual:
Character 145: '(' - Añadido a la pila. Pila actual: (
Character 146: ')' - Coincide con '('. Removido de la pila. Pila actual:
Character 148: '{' - Añadido a la pila. Pila actual: {
Character 265: '(' - Añadido a la pila. Pila actual: ({
Character 276: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Character 322: '(' - Añadido a la pila. Pila actual: ({
Character 375: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Character 435: '(' - Añadido a la pila. Pila actual: ({
Character 479: '(' - Añadido a la pila. Pila actual: ({(
Character 486: ')' - Coincide con '('. Removido de la pila. Pila actual: ({
Character 487: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Character 508: '}' - Coincide con '{'. Removido de la pila. Pila actual:
Resultado final: Balanceado
```

Captura ejecucion(Codigo en C que le falta la ultima llave de cierre) consola:

```
Archivo: "C:\Users\juani\OneDrive\Escritorio\Estructura-de-datos\tarea9\programa_suma.c"
Character 73: '(' - Añadido a la pila. Pila actual: (
Character 86: ')' - Coincide con '('. Removido de la pila. Pila actual:
Character 88: '{' - Añadido a la pila. Pila actual: {
Character 110: '}' - Coincide con '{'. Removido de la pila. Pila actual:
Character 145: '(' - Añadido a la pila. Pila actual: (
Character 146: ')' - Coincide con '('. Removido de la pila. Pila actual:
Character 148: '{' - Añadido a la pila. Pila actual: {
Character 265: '(' - Añadido a la pila. Pila actual: ({
Character 276: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Character 322: '(' - Añadido a la pila. Pila actual: ({
Character 375: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Character 435: '(' - Añadido a la pila. Pila actual: ({
Character 479: '(' - Añadido a la pila. Pila actual: ({(
Character 486: ')' - Coincide con '('. Removido de la pila. Pila actual: ({
Character 487: ')' - Coincide con '('. Removido de la pila. Pila actual: {
Error: Se esperaba un '}'
Resultado final: No balanceado
```