

# Reporte de Tarea: Menú de Opciones en Java

Programación Orientada a Objetos II

**Estudiante: Francisco López Velázquez**

Profesor: Dr. Carlos Hugo García Capulín

Tarea: 1 - Implementación de Menú de Opciones con Recursividad

Campus: Irapuato-Salamanca

Fecha de Entrega: Fecha de Entrega

# Introducción

Este reporte documenta la implementación de un programa en Java que permite al usuario elegir entre varias opciones de cálculo: sumatoria de números, factorial de un número, serie de Fibonacci y la solución de una ecuación de segundo grado. El código hace uso de funciones recursivas para las tres primeras opciones. Además, se incluye un manejo sencillo de ecuaciones de segundo grado, utilizando la fórmula general.

El objetivo de este ejercicio es reforzar conceptos de recursividad, estructura de control en Java y la organización de un programa a través de un menú interactivo. En este reporte se describe el código, las pruebas realizadas, y los resultados obtenidos.

## Desarrollo

El siguiente es el código del programa implementado en Java. El código incluye comentarios que explican las diferentes partes del mismo para facilitar su comprensión. Las funciones para la sumatoria, factorial y Fibonacci utilizan recursividad.

```
1 import java.util.Scanner;
2
3 public class MenuRecursivo {
4     public static void main(String[] args) {
5         // Creamos un scanner para capturar la entrada del usuario
6         Scanner scanner = new Scanner(System.in);
7         int opcion; // Aquí almacenaremos la opción seleccionada
8
9         do {
10             // Imprimimos el menú con las opciones disponibles
11             System.out.println("----- Menu de opciones: -----");
12             System.out.println("1. Sumatoria");
13             System.out.println("2. Factorial");
14             System.out.println("3. Serie de Fibonacci");
15             System.out.println("4. Solucion de ecuacion de segundo
16                                 grado");
17             System.out.println("5. Salir");
18             System.out.print("Elige una opcion: ");
19             opcion = scanner.nextInt(); // Leemos la opción
20             seleccionada por el usuario
21
22             // Dependiendo de la opción, llamamos a la función
23             correspondiente
24             switch (opcion) {
25                 case 1:
26                     System.out.print("Ingresa un numero para la
27                                     sumatoria: ");
28                     int numSum = scanner.nextInt();
29                     System.out.println("La sumatoria es: " +
30                                     sumatoria(numSum));
```

```

26         break;
27     case 2:
28         System.out.print("Ingresa un numero para el
29             factorial: ");
30         int numFact = scanner.nextInt();
31         System.out.println("El factorial es: " +
32             factorial(numFact));
33         break;
34     case 3:
35         System.out.print("Ingresa un numero para la
36             serie Fibonacci: ");
37         int numFib = scanner.nextInt();
38         System.out.println("El termino Fibonacci es: " +
39             fibonacci(numFib));
40         break;
41     case 4:
42         System.out.print("Ingresa coeficiente a: ");
43         double a = scanner.nextDouble();
44         System.out.print("Ingresa coeficiente b: ");
45         double b = scanner.nextDouble();
46         System.out.print("Ingresa coeficiente c: ");
47         double c = scanner.nextDouble();
48         resolverEcuacion(a, b, c);
49         break;
50     case 5:
51         System.out.println("Saliendo del programa...");
52         break;
53     default:
54         System.out.println("Opcion invalida.");
55     }
56 } while (opcion != 5); // Repetimos el men hasta que se
57     elija la opci n de salir
58 scanner.close(); // Cerramos el esc ner
59 }
60
61 // Funcion recursiva para la sumatoria de un n mero
62 public static int sumatoria(int n) {
63     if (n == 1) {
64         return 1; // Caso base: si es 1, regresamos 1
65     }
66     return n + sumatoria(n - 1); // Caso recursivo: n +
67         sumatoria de n-1
68 }
69
70 // Funcion recursiva para el factorial
71 public static int factorial(int n) {
72     if (n == 1) {

```

```

67         return 1; // Caso base
68     }
69     return n * factorial(n - 1); // Caso recursivo: n *
        factorial de n-1
70 }
71
72 // Funcion recursiva para la serie de Fibonacci
73 public static int fibonacci(int n) {
74     if (n == 0 || n == 1) {
75         return n; // Casos base: si es 0 o 1, regresamos el
        mismo n mero
76     }
77     return fibonacci(n - 1) + fibonacci(n - 2); // Caso
        recursivo
78 }
79
80 // Funcion para resolver ecuacion de segundo grado usando la
    f rmula general
81 public static void resolverEcuacion(double a, double b, double c
    ) {
82     double discriminante = b * b - 4 * a * c;
83     if (discriminante > 0) {
84         double x1 = (-b + Math.sqrt(discriminante)) / (2 * a);
85         double x2 = (-b - Math.sqrt(discriminante)) / (2 * a);
86         System.out.println("Las soluciones son: x1 = " + x1 + "
            y x2 = " + x2);
87     } else if (discriminante == 0) {
88         double x = -b / (2 * a);
89         System.out.println("La solucion es unica: x = " + x);
90     } else {
91         System.out.println("No existen soluciones reales.");
92     }
93 }
94 }

```

Listing 1: Código del programa en Java

```
1  import java.util.Scanner;
2
3  Codeium: Refactor | Explain
4  public class MenuRecursivo {
5      Codeium: Refactor | Explain | Generate Javadoc | X
6      public static void main(String[] args) {
7          // Creamos un escáner para capturar la entrada del usuario
8          Scanner scanner = new Scanner(System.in);
9          int opcion; // Aquí almacenaremos la opción seleccionada
10
11      do {
12          // Imprimimos el menú con las opciones disponibles
13          System.out.println("----- Menu de opciones: -----");
14          System.out.println("1. Sumatoria");
15          System.out.println("2. Factorial");
16          System.out.println("3. Serie de Fibonacci");
17          System.out.println("4. Solucion de ecuacion de segundo grado");
18
19          // Aquí se debería implementar la lógica para cada opción
20          // y actualizar la opción seleccionada.
21
22      } while (opcion != 5);
23
24      System.out.println("Programa terminado.");
25  }
```

PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS Filter (e.g. text, !exclude, \escape)

```
----- Menu de opciones: -----
1. Sumatoria
2. Factorial
3. Serie de Fibonacci
4. Solucion de ecuacion de segundo grado
5. Salir
Elige una opcion:
```

Figure 1: Código completo del programa en Java

## Pruebas y Resultados

A continuación, se describen las pruebas realizadas, los datos de entrada y los resultados obtenidos. Al final de cada prueba, se incluye un espacio para insertar una captura de pantalla de la ejecución en Visual Studio Code.

### Prueba 1: Sumatoria

Entrada: 5

Salida esperada: La sumatoria es: 15

Captura de pantalla:

### Prueba 2: Factorial

Entrada: 4

Salida esperada: El factorial es: 24

Captura de pantalla:

```
4. Solucion de ecuacion de segundo grado
5. Salir
Elige una opcion:
→ 1

Ingresa un numero para la sumatoria:
→ 5

La sumatoria es: 15
----- Menu de opciones: -----
> |
```

Figure 2: Prueba 1: Sumatoria

```
5. Salir
Elige una opcion:
→ 2

Ingresa un numero para el factorial:
→ 4

El factorial es: 24
----- Menu de opciones: -----
1. Sumatoria
>
```

Figure 3: Prueba 2: Factorial

### Prueba 3: Serie de Fibonacci

Entrada: 6

Salida esperada: El término Fibonacci es: 8  
6

Captura de pantalla:

```
3. Serie de Fibonacci
4. Solucion de ecuacion de segundo grado
5. Salir
Elige una opcion:
→ 3

Ingresa un numero para la serie Fibonacci:
→ 6

El termino Fibonacci es: 8
> |
```

Figure 4: Prueba 3: Serie de Fibonacci

#### Prueba 4: Solución de ecuación de segundo grado

**Entrada:**  $a = 1$ ,  $b = -3$ ,  $c = 2$

**Salida esperada:** Las soluciones son:  $x_1 = 2.0$  y  $x_2 = 1.0$

Captura de pantalla:

widthwidth

Figure 5: Prueba 4: Solución de ecuación de segundo grado

## Enlace al código fuente en GitHub

Puedes encontrar todo el código y recursos sobre este reporte en el siguiente enlace: Repositorio GitHub: Funciones Recursividad Java