



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorios de docencia

Laboratorio de Computación Salas A y B

Profesor(a): M.C. RENE ADRIAN DAVILA PEREZ

Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS

Grupo: 01

No de Práctica(s): 01 y 02

Integrante(s): 322118311

322094028

322092842

322078673

322067738

*No. de lista o
brigada:* 03

Semestre: 2026-1

Fecha de entrega: 29/08/2025

Observaciones:

CALIFICACIÓN: _____

Índice

1. Introducción	2
2. Marco Teórico	2
3. Desarrollo	3
4. Resultados	4
5. Conclusiones	6
6. Bibliografía	6

1. Introducción

- **Planteamiento del problema.** Para esta practica se realizara un programa en Java que mediante un menú, permita calcular el factorial de un numero haciendo uso de la recursividad, la serie de fibonacci o la serie de la conjetura de Collatz, segun lo indique el usuario.
- **Motivación.** Problemas matemáticos como el cálculo del factorial, la serie de Fibonacci y la conjetura de Collatz son perfectos para aplicar la recursividad y como combinarla con estructuras de control.
- **Objetivos.** Al darle solución al problema esperamos fortalecer nuestras bases en el lenguaje Java mediante la aplicación de estructuras de control y el análisis del funcionamiento de la recursividad, valorando su eficacia en la resolución de diferentes tipos de problemas numéricos.

2. Marco Teórico

Factorial de un numero: El factorial de un número se refiere a los productos de los numeros enteros positivos que le anteceden a un número n . [1]

$$n! = \begin{cases} 1, & \text{si } n = 0, \\ n \cdot (n - 1)!, & \text{si } n > 0. \end{cases}$$

Serie de Fibonacci: Para la serie de Fibonacci no es mas que dados los números 0 y 1, cada número de la serie es sencillamente la suma de sus dos numeros inmediatos que le anteceden. [2]

$$F(n) = \begin{cases} 0, & \text{si } n = 0, \\ 1, & \text{si } n = 1, \\ F(n - 1) + F(n - 2), & \text{si } n > 1. \end{cases}$$

Conjetura de Collatz: Esta conjetura dice que eligas el numero que sea siempre se puede llegar a 1 dividiendo por 2, si el algun momento te encuentras un numero impar se le suma 1 y se vuelve a dividir hasta llegar a 1, esta conjetura aplica para todos los numeros R e incluso aplica para los complejos. [3]

$$C(n) = \begin{cases} 1, & \text{si } n = 1, \\ C\left(\frac{n}{2}\right), & \text{si } n \text{ es par}, \\ C(3n + 1), & \text{si } n \text{ es impar y } n > 1. \end{cases}$$

Recursividad: Por un lado tenemos la sección en la que la función se llama a sí misma haciendo el problema cada vez mas pequeño. Por otro lado, tiene que existir siempre una condición en la que la función retorna sin volver a llamarse (caso base). Es muy importante porque de lo contrario, la función se llamaría de manera indefinida.[4]

$$F(n) = \begin{cases} \text{Valor conocido,} & \text{si se cumple la condición de terminación (caso base),} \\ f(n, g(n')), & \text{en otro caso (paso recursivo).} \end{cases}$$

3. Desarrollo

Para iniciar con la resolución de problema comenzaremos importando la libreria Scanner, la cual servira para que el programa pueda leer datos de entrada proporcionados por el usuario, posteriormente declaramos la clase principal la cual llamaremos "Practical1" dentro de ella definiremos los siguientes metodos.

Static long factorial: Recibira un valor de tipo entero proporcionado por el usuario, para posteriormente entrar a una estructura de control if-else y comparar el valor dado, en caso de ser igual a 0 o 1, nos regresara 1, de lo contrario, mediante el uso de la recursividad realizara la siguiente multiplicacion: **n*factorial(n-1)**

Static long fibonacci: Recibira un valor de tipo entero para posteriormente con una estructura if-else verificar si el valor proporcionado por el usuario es igual a 0, de ser el caso, el metodo nos regresara 0, en otro caso si el valor proporcionado es igual a 1, el metodo nos regresara 1, de no cumplirse los casos base, haciendo uso de la recursividad, el programa realizara la siguiente operacion **fibonacci(n-1)+fibonacci(n-2)**.

Static long Collatz: Recibira un valor de tipo entero proporcionado por el usuario y mediante un ciclo while, que mientras el valor sea diferente de 1, entrara en una estructura if-else que verificara si el numero es par, de ser asi lo dividira entre 2, en caso de ser impar lo multiplicara por 3 y le sumara 1, finalmente ira imprimiendo la secuencia.

Metodo main:

Por ultimo crearemos el menú en donde usaremos un objeto Scanner para leer la opcion que introduzca el usuario, con un ciclo do-while imprimiremos las opciones del programa para posteriormente preguntar al usuario la opcion a elegir y mediante una sentencia de control switch se llamara a cada metodo, de ser el caso 1, llamaremos al metodo factorial, de ser el caso 2 entrara el metodo fibonacci, para el caso 3 el metodo sera el de Collatz, finalmente para el caso 4 sera la opcion de salida. todo esto mientras la opción sea diferente de 4.

4. Resultados

```
MENU
-----

1.FACTORIAL DE UN NUMERO
2.CALCULAR LA SERIE DE FIBONACCI
3.CONJETURA DE COLLATZ
4.SALIR

INTRODUCE LA OPCION
█
```

Figura 1: Menú

En este captura podemos observar el menú asociado a las diferentes opciones en el programa.

```
1.FACTORIAL DE UN NUMERO
2.CALCULAR LA SERIE DE FIBONACCI
3.CONJETURA DE COLLATZ
4.SALIR

INTRODUCE LA OPCION
1
INGRESA EL FACTORIAL QUE QUIERES CALCULAR
5
El factorial de dicho numero es 120
```

Figura 2: Opción Factorial

El usuario al ingresar la opcion 2 introduce el numero, el programara imprimira el factorial de dicho numero.

```
MENU
-----

1.FACTORIAL DE UN NUMERO
2.CALCULAR LA SERIE DE FIBONACCI
3.CONJETURA DE COLLATZ
4.SALIR

INTRODUCE LA OPCION
2
INGRESA EL NUMERO DE LA SERIE FIBONACCI QUE QUIERES CALCULAR
5
El número de la serie es :
0,1,1,2,3,
```

Figura 3: Opción Serie de Fibonacci

Después de que el usuario ingrese el número, el programa imprimirá la serie de fibonacci hasta el término asociado al número proporcionado por el usuario.

```
MENU
-----

1.FACTORIAL DE UN NUMERO
2.CALCULAR LA SERIE DE FIBONACCI
3.CONJETURA DE COLLATZ
4.SALIR

INTRODUCE LA OPCION
3
INGRESA EL NUMERO DE LA SERIE DE COLLATZ QUE QUIERES CALCULAR
50
25 76 38 19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

Figura 4: Opción Conjetura de Collatz

Para el número ingresado al usuario, el programa calculará los elementos de la serie asociados a la conjetura de Collatz.

```
1.FACTORIAL DE UN NUMERO
2.CALCULAR LA SERIE DE FIBONACCI
3.CONJETURA DE COLLATZ
4.SALIR

INTRODUCE LA OPCION
4
bye bye
```

Figura 5: Opción salir

Finalmente luego de introducir la opcion 4, el programa termina.

5. Conclusiones

Consideramos que logramos alcanzar los objetivos, ya que a lo largo del desarrollo de esta programa en Java, logramos aplicar correctamente las estructuras de control y pusimos en practica el concepto de recursividad en la funciones que implementamos para realizar los calculos que ofrece el programa, nos permitio ademas analizar la eficacia de la recursividad. De esta manera ampliamos nuestras bases de programación en lenguaje Java y fortaleció nuestra compresion de el concepto de recursividad.

6. Bibliografia

[1] J. de León Razo, "Cómputo Aplicado," Unidades de Apoyo para el Aprendizaje, CUAED/Facultad de Ingeniería-UNAM, REA, 2019.

[2] M. E. (Autor), "Definición recursiva de la serie de Fibonacci,"trabajo académico no publicado, UNAM, 2025.

[3] "La conjetura de Collatz,"YouTube, 2025.
[<https://youtu.be/HpcYW08Ug7g?si=qxy7l6zsNXT2LgGJ>]

[4] El Libro de Python, «Recursividad», [ellibrodepython.com](https://ellibrodepython.com/recursividad).
<https://ellibrodepython.com/recursividad> (consultado el 29 de agosto de 2025).