



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorios de docencia

Laboratorio de Computación Salas A y B

Profesor(a): Dávila Pérez René Adrián

Asignatura: Programación Orientada a Objetos

Grupo: 1

No de Práctica(s): 1

Integrante(s): 322089020

322089817

322151194

425091586

322085390

*No. de lista o
brigada:* Equipo 4

Semestre: 2026-1

Fecha de entrega: 29 de agosto de 2025

Observaciones:

CALIFICACIÓN: _____

Índice

1. Introducción	2
2. Marco Teórico	3
3. Desarrollo	6
4. Resultados	8
5. Conclusiones	10

1. Introducción

Planteamiento del Problema

1. Cálculo del factorial de un número

El algoritmo debe mandar un numero a una función la cual mediante recursividad calcule su factorial, una vez calculado el resultado lo muestre al usuario.

2. Generación de la serie de Fibonacci

Se debe invocar una función la cual calcule toda la serie ordenada hasta el indice indicado, y la imprima en orden en la terminal.

3. Resolución de la Conjetura de Collatz

Iniciar la Conjetura de Collatz desde un numero n, cada calculo que realice se imprimirá en la terminal para mostrar el camino que se tomo para llegar al caso base.

Motivación

La práctica ayudara a mostrar de forma gráfica y experimental los conocimientos teóricos de programación en Java orientada a objetos aprendidos hasta el momento, permitiendonos observar cómo se aplican los conceptos y herramientas tales como la recursividad. Se realizarán ejercicios que permiten ver la implementación de estructuras de control y llamados a funciones, mostrando paso a paso cómo se resuelven los problemas mediante algoritmos y lógicas bien estructuradas. De esta manera, podremos entender de mejor forma, cómo funciona cada estructura y cada paso del algoritmo para llegar al resultado final, haciendo más evidente la relación entre la teoría y la práctica en la programación.

Objetivos

1. Realizar un código interactivo con el usuario que le permita seleccionar el cálculo que desea realizar y que se ejecute en bucle hasta que se seleccione la opción de salir.
2. Realizar un código que calcule el factorial de un número n utilizando la recursividad como herramienta hasta llegar al caso base (1).
3. Implementar una función que imprima la serie de Fibonacci hasta el número que indique el índice que el usuario ingresó.
4. Crear una función que permita visualizar el proceso de Collatz para cualquier número.

2. Marco Teórico

Variables

Las variables en Java son de tipo primitivo, es decir, siempre ocupan el mismo número de bytes en memoria, sin importar el valor que almacenen. Estas características permiten una gestión más eficiente de los recursos y un control preciso sobre el manejo de datos en los programas. [1]

Funciones

Las funciones y los procedimientos constituyen un recurso esencial dentro de la programación, en particular en el lenguaje Java. Este lenguaje ofrece la posibilidad de definir métodos propios y utilizarlos de manera sencilla, además de brindar acceso a una amplia variedad de métodos disponibles en sus librerías. Estos métodos abarcan desde operaciones matemáticas, manejo de archivos, hasta manipulación de fechas, entre otros.

El uso de funciones permite automatizar tareas recurrentes y, gracias al empleo de parámetros o argumentos, es posible generalizarlas para adaptarlas a diferentes contextos. En

términos sintácticos, la declaración de una función implica definir el nivel de acceso, modificadores, tipo de retorno, nombre de la función y, en caso necesario, los parámetros que recibirá.

Librerías o bibliotecas

En Java, al igual que en muchos otros lenguajes de programación, una librería puede entenderse como un conjunto de clases que incluyen diversos métodos y atributos. Su principal utilidad radica en que simplifican numerosas operaciones, ya que permiten reutilizar código previamente desarrollado. De esta manera, el programador aprovecha las clases, métodos y atributos que ofrecen estas librerías, evitando tener que implementarlos desde cero.

El uso de librerías, tanto propias como externas, es muy sencillo. Para importar librerías en Java se emplea la palabra clave *import*, seguida de la ruta del paquete o clase que se desee incorporar al proyecto. Cabe resaltar que esta instrucción permite agregar una clase en particular o un conjunto de ellas (paquete) según las necesidades del programa.

Factorial

El factorial es una operación representada por la notación $n!$, que consiste en el producto de todos los números enteros positivos menores o iguales a n .

Por ejemplo, el factorial de siete se expresa como:

$$7! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 = 5040$$

Esta operación es ampliamente utilizada en combinatoria, probabilidad y análisis de algoritmos, siendo fundamental para resolver problemas de conteo y organización de elementos.

[5]

Serie de Fibonacci

La serie de Fibonacci, atribuida al matemático italiano Leonardo de Pisa, mejor conocido como Fibonacci, se construye a partir de dos primeros números iniciales. A partir de ellos, cada término siguiente resulta de la suma de los dos inmediatamente anteriores. [4]

Un aspecto interesante de esta serie es su manifestación en la naturaleza: aparece en la disposición de hojas, caracolas, patrones de crecimiento de ramas y en la reproducción de poblaciones. Asimismo, la serie está estrechamente relacionada con la proporción áurea, la cual ha sido aprovechada en disciplinas como el arte, la arquitectura, las finanzas y el diseño, donde se emplea para modelar procesos de crecimiento, predecir tendencias y crear composiciones armónicas. [2]

Conjetura de Collatz

La conjetura de Collatz es un enigma matemático que parte de cualquier número entero mayor que uno. Su regla establece que:

- Si el número es par, se divide entre dos.
- Si el número es impar, se multiplica por tres y se suma uno.

Este proceso se repite sucesivamente, y la conjetura afirma que, sin importar el número inicial, siempre se llegará al valor de uno. Aunque es sencillo de enunciar, el comportamiento de la secuencia depende de la magnitud y complejidad del número inicial, y hasta la fecha no ha sido formalmente demostrada para todos los enteros. [3]

3. Desarrollo

Implementación del Código

Es indispensable para la ejecución del programa el uso de un menú y la clase *Scanner* para recibir datos de entrada, primero el usuario ingresa que algoritmo usará, posteriormente con un *switch* se ejecuta el código correspondiente, en los primeros 3 casos en los que el *switch* puede caer, primero se pide el valor de n que es el número o la posición necesario para ejecutar el algoritmo correspondiente en el caso de que el *switch* caiga en el caso 4 el programa finaliza y si el usuario ingresa una opción diferente se informa que la opción no es válida, todo esto se repite dentro de un *while* a menos que la opción sea 4 el *while* se rompe.

Funciones:

- **Factorial:** Recibe como parámetro el número para calcular su factorial, dado que el factorial de un número n , es n multiplicado por el factorial del n menos uno ($n! = n * (n - 1)!$). Esto nos dice que la función para calcular el factorial de n puede ser recursiva. Contemplando el caso base que es cuando $n = 0$, este es el primer paso, un *if*, y en el caso dado que $n > 0$, la función devuelve por definición $n * factorial(n - 1)$.
- **Fibonacci:** Recibe como parámetro la posición en la serie de Fibonacci que se debe mostrar. En el caso de la serie de Fibonacci hay 2 casos base para este algoritmo cuando $n = 0$ y $n = 1$, estas son las primeras estructuras que se necesitan, en caso de que $n > 1$, la función devuelve por definición $fibonacci(n - 1) + fibonacci(n - 2)$ que es la fórmula para obtener el n -ésimo término de la serie de Fibonacci.
- **Mostrar Serie de Fibonacci:** Recibe como parámetro hasta que posición de la serie debe mostrar. Con un ciclo *for* iteramos desde 0 hasta la posición n de la cual queremos mostrar el número de la serie con esto podemos mostrar la serie en orden.
- **Conjetura de Collatz:** Recibe como parámetro el número para desarrollar la conje-

tura de Collatz esta consiste en tomar un número n y dependiendo si es par o impar se divide entre 2 o multiplica por 3 y se le suma 1, el fin de la conjetura es cuando $n = 1$, por lo tanto podemos considerar esto como el caso base, y donde entra *if* si se encuentra que $n = 1$ el programa finaliza, si n es par se llama a la misma función pero con el valor dividido en 2 como la definición, si n es impar se llama a función misma pero multiplicado por 3 más 1 por definición, esta función si bien no regresa ningún valor siempre que la función se ejecuta hasta que otra cosa, se muestra el valor de n .

Pruebas

Input(Factorial) : 1

Output(Factorial) : 1

Input(Fibonacci) : 10

Output(Fibonacci) : 0, 1, 1, 2, 3, 5, 8, 13, 21, 34

Input(Collatz) : 15

Output(Collatz) : 15, 46, 23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5, 16, 8, 4, 2, 1

4. Resultados

```
carlosdanc1@HP15-Daniel:~/daniel/POO/Practicas/Practica1_2$ javac Practica1_2.java
carlosdanc1@HP15-Daniel:~/daniel/POO/Practicas/Practica1_2$ java Practica1_2

Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 
```

Figura 1: Se ejecuta el programa y se muestra el menú.

```
Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 1
Introduzce el valor de n: 8
Factorial de 8 es 40320

Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 
```

Figura 2: Se selecciona la opción 1 (factorial) y se introduce el valor 'n' a calcular; se vuelve a mostrar el menú.

```
Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 2
Introduzce el valor de n:
6
0, 1, 1, 2, 3, 5,
```

Figura 3: La segunda opción devuelve la serie de Fibonacci de un cierto 'n' solicitado al usuario.

```
Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 3
Introduzce el valor de n:
5
5, 16, 8, 4, 2, 1,
```

Figura 4: La opción 3 imprime la conjetura de collartz a partir de el entero 'n' solicitado.

```
Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 9
Opcion invalida...

Practica 1
Menu (Seleccione una opcion):
1.- Factorial
2.- Serie de Fibonacci
3.- Conjetura de Collatz
4.- Salir
-> 4
Fin del programa...
carlosdanc1@HP15-Daniel:~/daniel/P00/Practicas/Practica1_2$
```

Figura 5: Si el usuario ingresa cualquier otra opción, el programa se lo indica y vuelve a mostrar el menú; si selecciona 4, el programa cierra.

5. Conclusiones

Al analizar los resultados se observa que la práctica cumplió con los objetivos propuestos, logrando desarrollar las lógicas y algoritmos necesarios para resolver los tres problemas, así como el menú interactivo de selección. La utilización de recursión en los códigos permitió optimizar las funciones, haciéndolas más compactas y eficientes, evitando redundancias y facilitando la lectura del programa. Asimismo, la correcta validación de las salidas en cada uno de los algoritmos permitió confirmar que la implementación se ajustó a la solución teórica planteada, garantizando resultados precisos y consistentes. En conjunto, la práctica evidenció que se logró aplicar de manera adecuada los conceptos vistos, integrando teoría y programación en un proyecto funcional y bien estructurado.

Referencias

- [1] P Pablo Garrido Abenza. *Comenzando a programar con JAVA*. Universidad Miguel Hernández, 2015.
- [2] Equipo de redacción de Drew. *Secuencia de Fibonacci*. Blog.wearedrew.co. Jul. de 2022. URL: <https://blog.wearedrew.co/concepts/que-es-la-secuencia-de-fibonacci>.
- [3] Víctor M. Manero y Alfonso Jesús Población. *La conjetura de Collatz: un problema "sencillo" que desafiará tu intuición*. abc.es. Enero 18 de 2021. URL: https://www.abc.es/ciencia/abci-conjetura-collatz-problema-sencillo-desafiara-intuicion-202101181439_noticia.html.
- [4] Jardín de los Matemáticos. *Sucesión Fibonacci*. <https://www2.ual.es/jardinmatema/sucesion-de-fibonacci/>. Sin fecha.
- [5] *Resumen de Factorial*. <https://teachy.ai/es/resumenes/educacion-secundaria/secundaria-2-grado/matematicas-a-espanol/factorial-or-resumen-tradicional-37359>. Accedido el 28 de agosto de 2025.