

## Actividad | 1 Algoritmos

### Introducción al Desarrollo de Software

Ingeniería en Desarrollo de Software

---



academias global

TUTOR: Sandra Luz Lara Devora.

---

ALUMNO: Aldair Quiñonez González.

---

FECHA: 15 de Abril del 2025

---

## Índice

Índice.....	1
1    Introducción .....	2
2    Descripción .....	2
3    Justificación .....	3
4    Desarrollo.....	4
4.1    Punto 1 .....	6
4.2    Punto 2 .....	7
5    Conclusión .....	9
6    Referencias.....	10

## **1 Introducción**

En esta actividad se desarrollan y analizan tres algoritmos fundamentales: uno para determinar si un número es primo otro para invertir un número de cuatro cifras y uno para descifrar los numero pares e impares. El propósito principal es fortalecer la lógica de programación, el uso de estructuras de control y la correcta aplicación de operaciones matemáticas básicas. El primer algoritmo permite verificar si un número es primo, es decir, si solo es divisible entre sí mismo y el uno. Para lograrlo, se utiliza un contador de divisores que incrementa cada vez que el número es divisible sin residuo. Al finalizar, si el número tiene exactamente dos divisores, se concluye que es primo. El segundo algoritmo está diseñado para invertir un número de cuatro dígitos. Mediante divisiones sucesivas y el uso del operador módulo, se extraen los dígitos del número original y se reorganizan en orden inverso como por tercero está el que sirve para identificar si un número es par o impar. Estos algoritmos no solo refuerzan el pensamiento lógico, sino que también permiten entender mejor el funcionamiento interno de estructuras como ciclos, condicionales y operadores aritméticos. Esta actividad es clave para desarrollar habilidades en la resolución de problemas mediante programación estructurada.

## **2 Descripción**

Consiste en el desarrollo de tres algoritmos básicos que permiten fortalecer la lógica de programación mediante el uso de estructuras condicionales y repetitivas. Cada algoritmo cumple una función específica orientada a resolver distintos problemas utilizando operaciones matemáticas y control de flujo. El primer algoritmo se encarga de verificar si un número

ingresado por el usuario es primo. Para ello, evalúa cuántas veces es divisible exactamente y, si solo es divisible entre sí mismo y el uno, determina que se trata de un número primo. El segundo algoritmo permite invertir un número de cuatro cifras. Mediante el uso del operador módulo y divisiones sucesivas, se extraen los dígitos del número original para reordenarlos en forma inversa, mostrando como resultado final el número invertido. Por último, el tercer algoritmo, llamado par e impar, identifica si un número es par o impar. Su objetivo es mostrar únicamente los números impares dentro de un rango determinado. Este algoritmo refuerza el uso de condicionales al evaluar el residuo de la división del número entre 2, y demuestra cómo se pueden aplicar filtros en secuencias numéricas.

### **3 Justificación**

La presente actividad tiene como objetivo principal el fortalecimiento de habilidades lógico-matemáticas y de razonamiento mediante la implementación de algoritmos básicos en pseudocódigo. A través del diseño y análisis de tres algoritmos —número primo, inversión de número y par/impar—, el estudiante tiene la oportunidad de aplicar estructuras de control como condicionales y ciclos repetitivos, que son pilares fundamentales en la programación estructurada. La construcción de estos algoritmos permite al alumno comprender de forma práctica cómo se resuelven problemas cotidianos mediante el uso de instrucciones secuenciales y decisiones lógicas. El algoritmo para verificar si un número es primo, por ejemplo, introduce el concepto de divisibilidad, mientras que el de invertir un número refuerza el manejo de operaciones aritméticas y el almacenamiento de datos intermedios. Por su parte, el algoritmo “par e impar” permite identificar los números impares dentro de un rango, ayudando a reconocer patrones numéricos y aplicar condiciones de forma eficiente. Esta actividad también fomenta la

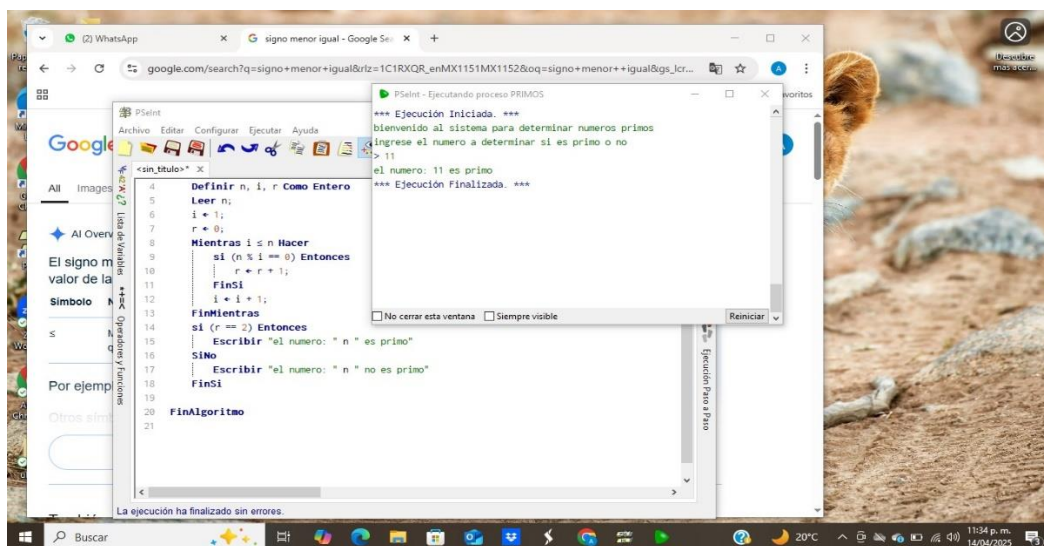
atención al detalle, el análisis de errores y la mejora continua del código. En conjunto, los ejercicios presentados contribuyen al desarrollo de competencias necesarias para resolver problemas de forma lógica, ordenada y eficiente, sentando así las bases para enfrentar retos más complejos en el área de programación.

#### **4 Desarrollo**

Algoritmo 1: Determinar si un número es primo;

1.      Mostrar un mensaje de bienvenida para que el usuario sepa lo que hará el programa.
2.      Solicitar al usuario que ingrese un número entero. Este será el número que se evaluará.
3.      Definir las variables necesarias: una para contar divisores y otra para hacer divisiones.
4.      Inicializar las variables necesarias (por ejemplo, el contador en 0 y el divisor en 1).
5.      Usar una estructura repetitiva para evaluar cuántas veces el número puede ser dividido exactamente (sin residuo).

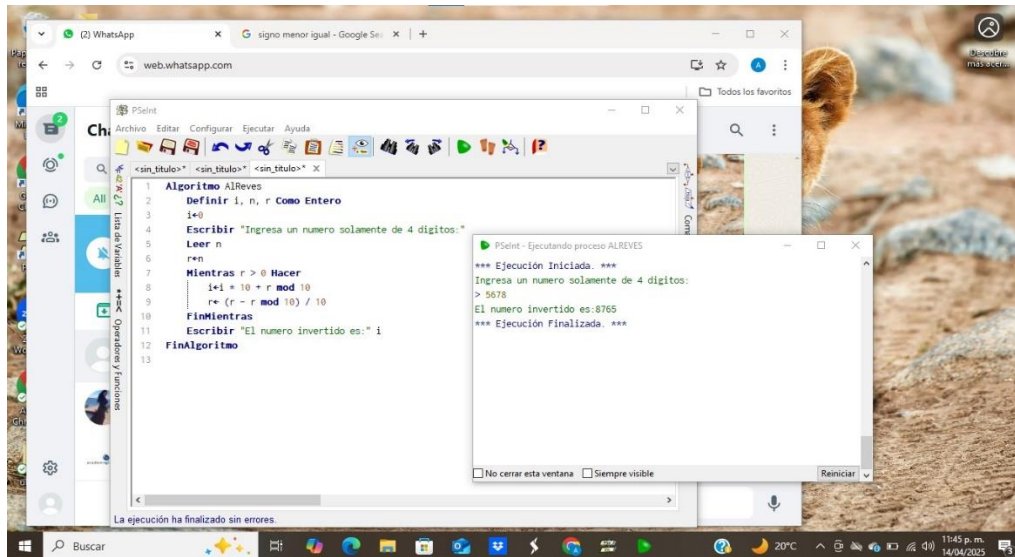
6. Por cada número desde 1 hasta el valor ingresado, verificar si el residuo de la división es 0.
7. Si el residuo es 0, aumentar el contador de divisores en 1.
8. Al finalizar el ciclo, analizar el valor del contador:
  - Si el contador es igual a 2, el número es primo (solo se divide entre 1 y él mismo).
  - Si el contador es mayor a 2, el número no es primo (tiene más divisores).
9. Mostrar un mensaje al usuario indicando si el número ingresado es o no primo.
10. Terminar el programa con un mensaje de cierre.



#### 4.1 Punto 1

Algoritmo 2: Invertir un número de 4 dígitos;

1. Solicitar al usuario que escriba un número de cuatro cifras mediante un mensaje en pantalla.
2. Dividir el número entre 10 y guardar el residuo. Este residuo corresponde al último dígito.
3. Guardar ese residuo y dividir el cociente nuevamente entre 10. Multiplicar el residuo anterior por 10 y sumarle el nuevo.
4. Repetir el paso anterior dos veces más, siempre multiplicando el resultado parcial por 10 y sumándole el siguiente residuo.
5. Al finalizar, se tendrá el número completamente invertido.
6. Mostrar el número invertido al usuario con un mensaje que diga, por ejemplo: “El número invertido es: \_\_\_”.
7. Terminar el programa con un mensaje de despedida.



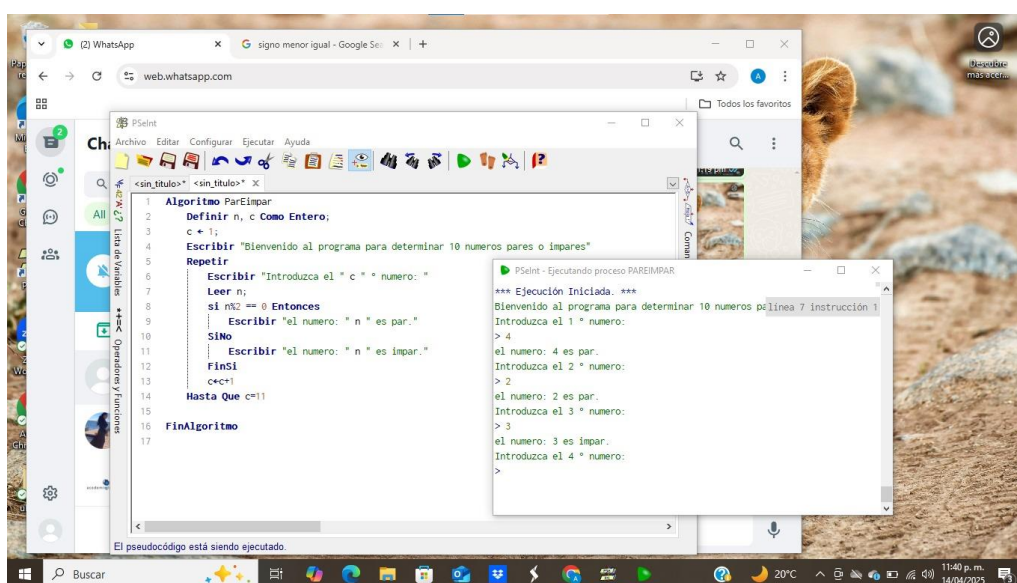
## 4.2 Punto 2

Algoritmo 3: Par e Impar;

1. “Dar un mensaje de bienvenida al usuario que quiera ingresar a este algoritmo.
2. Solicitar los 10 números requeridos con el siguiente dialogo: Proporcione el número (número que se quiere obtener );, hasta llegar a los 10 números solicitados.
3. Inicializar la variable divisor a 2 para determinar el residuo.
4. Hacer la operación de división de cada número y si el residuo es 0 se imprimirá un diálogo que sería: el numero (número dividido) es par.



5. De lo contrario si el residuo es diferente de 0 se imprimirá el siguiente diálogo: en numero (número dividido) es impar.
6. Así se continuará hasta dividir los 10 números proporcionados.
7. Se finalizará con un mensaje de despedida.” Domingo tutor, 2025.



## 5 Conclusión

A través del desarrollo de esta actividad se logró reforzar el conocimiento sobre la construcción de algoritmos utilizando pseudocódigo, aplicando correctamente estructuras de control como condicionales y ciclos repetitivos. Los tres algoritmos desarrollados —determinación de números primos, inversión de un número de cuatro cifras y el algoritmo par e impar— permitieron poner en práctica conceptos fundamentales de lógica matemática y programación estructurada. Cada uno de los algoritmos abordó un problema diferente, pero con un enfoque común: la correcta solución mediante pasos secuenciales, análisis lógico y evaluación de condiciones. Esta práctica no solo ayudó a comprender cómo se procesan los datos en un algoritmo, sino también la importancia de la precisión en la lógica utilizada. Además, se fortalecieron habilidades como el razonamiento lógico, la resolución de problemas y la capacidad de traducir ideas matemáticas en instrucciones claras para un programa. En resumen, esta actividad fue clave para adquirir una base sólida que servirá para enfrentar desafíos más complejos en futuros desarrollos de software.

## 6 Referencias

PSeInt.com

Tutorías con Domingo

[https://drive.google.com/file/d/1NUc39qf3R5RuVhWcn2eRqjuTV73EoGp\\_/view?usp=drivesdk](https://drive.google.com/file/d/1NUc39qf3R5RuVhWcn2eRqjuTV73EoGp_/view?usp=drivesdk)

<https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/politicas-y-gestion/entender-son-funcionan-algoritmos>