LABORATORIO 1 (Septiembre de 2025)

# Lozano G. Deicy F., Hernandez O. Johan E.

**Resumen - Este documento presenta la resolución de diferentes problemas mediante algoritmos implementados en Python. Cada ejercicio aborda un contexto distinto: el cálculo de retrasos en proyectos de software (Algoritmo A), el volumen de una esfera en procesos de manufactura (Algoritmo B), la autenticación de usuarios en sistemas de seguridad (Algoritmo C) y la aplicación de descuentos en compras en línea (Algoritmo D).**

**El trabajo demuestra cómo los algoritmos permiten transformar situaciones reales en soluciones lógicas y programables, facilitando la optimización de procesos, el análisis de información y la toma de decisiones.**

**Abstract - This document presents the resolution of different problems using algorithms implemented in Python. Each exercise addresses a different context: calculating delays in software projects (algorithm A), the volume of a sphere in manufacturing processes (algorithm B), user authentication in security systems (algorithm C), and applying discounts to online purchases (algorithm D).**

**The work demonstrates how algorithms can transform real-world situations into logical and programmable solutions, facilitating process optimization, information analysis, and decision-making.**

# introducción

En el presente documento se plantean y resuelven diferentes situaciones mediante la aplicación de la metodología de resolución de problemas y el uso de algoritmos en Python. Cada uno de los casos expuestos representa una problemática real en diversos contextos: la gestión de proyectos de software, los cálculos geométricos aplicados a la manufactura, los sistemas de autenticación y control de acceso, y la administración de descuentos en un entorno de comercio electrónico.

El propósito principal es mostrar cómo los algoritmos permiten transformar requerimientos prácticos en soluciones lógicas y estructuradas, que posteriormente se implementan en programas computacionales. A través de este proceso, se fortalecen competencias clave como el análisis, la abstracción y la programación, fundamentales en el área de ingeniería de sistemas.

De esta manera, los ejercicios presentados no solo contribuyen a la comprensión de los conceptos teóricos, sino que también ofrecen ejemplos aplicados en distintos ámbitos, demostrando la importancia de la programación para resolver problemas cotidianos, optimizar procesos y apoyar la toma de decisiones.

Objetivo general

Marco teórico

**Algoritmo A :** En la gestión de proyectos de desarrollo de software, la entrega puntual de tareas es un factor crítico para el éxito. Sin embargo, los proyectos suelen estar sujetos a cambios en requerimientos, prioridades y recursos, lo que genera incertidumbre en la planificación y ejecución. Estos cambios pueden provocar retrasos en tareas específicas, afectando no solo los cronogramas locales, sino también la entrega global del producto final.

Ayuda en la gestión de proyectos: Identifica cuánto se ha atrasado un proyecto respecto al cronograma.

Mejora la eficiencia: Ayuda a analizar cuellos de botella operativos.

Afecta la toma de decisiones empresariales: ayuda a las empresas a tomar medidas correctivas para reducir retrasos.

Útil en logística y cadena de suministro: Garantiza entregas oportunas e identifica posibles ineficiencias.

Aunque estas métricas son útiles, por sí solas no permiten **predecir tendencias ni adaptarse a cambios continuos**, lo cual representa una limitación significativa en proyectos ágiles.

**Cuello de botella operativo es** cualquier restricción o punto de un proceso que limita la capacidad de rendimiento de todo el sistema, ralentizando el flujo de trabajo, causando retrasos y afectando la eficiencia general.

(Javed, n.d.)

**Algoritmo B :** El laboratorio consta de dos pasos principales: leer el radio de la esfera y luego calcular el volumen utilizando la fórmula V = (4.0/3.0) \_ π \_ r³. Crearás un programa sencillo que solicite al usuario que ingrese el radio, calcule el volumen y muestre el resultado. Al final de este laboratorio, tendrás una mejor comprensión de los cálculos geométricos.

Primero, declarar una variable para almacenar el radio y solicitar al usuario que ingrese el valor. Luego, calcular el volumen de la esfera utilizando la fórmula matemática y el valor constante de π. Finalmente, imprimir el volumen en la consola.

(labex, 2017)

**Algoritmo C:** los usuarios pueden cambiar sus contraseñas libremente, y el sistema remoto no necesita el directorio de contraseñas ni las tablas de verificación para autenticar a los usuarios. Una vez configurado el entorno de red seguro, la autenticación puede ser gestionada únicamente por ambas partes involucradas.

(Shieh & Yang, 1999, #)

**Algoritmo D:** un sistema de gestión de cupones digitales que busca desmaterializar el proceso de emisión de cupones desde su origen hasta su estado final , pasando por el vendedor del producto, lo que permite reducir costos, prevenir fraudes, mejorar la eficiencia de los procedimientos y conocer los patrones de uso de cupones por perfil de consumidor.

Proporciona control de calidad de los cupones desmaterializados y acceso móvil a los consumidores, permitiendo la conexión e integración con los sistemas de información de emisores y vendedores, lo que lo convierte en un sistema innovador con un alto grado de portabilidad de sus componentes.

**(Araújo, 2015, #)**

# DESARROLLO

# Análisis de los problemas a solucionar

## Problema de algoritmo A:

Descripción: La empresa de desarrollo de software necesita una herramienta que permita a los gerentes de proyecto calcular el retraso en días de las tareas que se entregan después de la fecha límite, y además, determinar el porcentaje de retraso con respecto al tiempo total asignado a cada tarea.

## Datos que el sistema requiere:

## Tiempo asignado a la tarea (en días).

## Días realmente utilizados para completarla.

## Proceso:

## Comparar el tiempo real con el tiempo asignado.

## Si el tiempo real es mayor que el asignado → existe retraso.

## Retraso en días = tiempo\_real – tiempo\_asignado.

## Porcentaje de retraso = (retraso / tiempo\_asignado) × 100.

## Si el tiempo real es menor o igual al asignado - no hay retraso.

## Resultado:

## Número de días de retraso.

## Porcentaje de retraso.

## Mensaje indicando si la tarea estuvo dentro del plazo o fuera de él.

## Problema de algoritmo B:

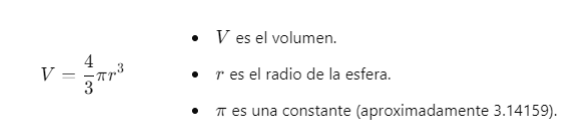
Descripción: La empresa de manufactura de globos y pelotas necesita conocer cuánto material se requiere para fabricar una pelota inflable.El material depende del volumen de la esfera, el cual se calcula a partir de su radio.Diseñar un programa que, dado el radio de la pelota, calcule el volumen de la esfera.

* Entrada de datos:

Se necesita ingresar el valor del radio de la pelota (un número positivo).

* Proceso:

Para hallar el volumen se utiliza la fórmula matemática de la esfera:



*Figura 1: explica la fórmula que se va a utilizar para la resolución del algoritmo B, fuente propia*

* Salida de datos:

El programa mostrará como resultado el volumen de la pelota, expresado en unidades cúbicas.

* Condiciones:

El radio ingresado debe ser mayor que cero, y el programa debe ser capaz de calcular el volumen para cualquier valor válido del radio.

## Problema de algoritmo C:

Descripción: La empresa de tecnología necesita un sistema que permita o deniegue el acceso de usuarios a un edificio de alta seguridad. Cada usuario cuenta con un nivel de acceso, una tarjeta de identificación y un registro de cambio de contraseña. El sistema debe evaluar esas condiciones y decidir si se autoriza o no el ingreso.

* Datos que el sistema necesita:

Nivel de acceso del usuario (número entre 0 y 5).

Nivel de acceso requerido para el área (número entre 1 y 5).

Estado de la tarjeta (activa o inactiva ).

Días desde el último cambio de contraseña .

* Proceso :

Comparar el nivel de acceso del usuario con el nivel requerido.

Verificar que la tarjeta esté activa.

Revisar si la contraseña fue cambiada en los últimos 30 días.

Si todas las condiciones son verdaderas, el sistema permitirá el acceso; de lo contrario, lo negará.

* Resultado:

Mostrar un mensaje:

Acceso permitido: si cumple todas las condiciones.

Acceso denegado: en caso contrario.

## Problema de algoritmo D:

Descripción: El departamento de TI de una tienda en línea necesita automatizar el cálculo de los descuentos aplicados a los clientes según distintas condiciones: monto de compra, membresía VIP y uso de código especial. El objetivo es desarrollar un algoritmo que determine el total a pagar después de aplicar los descuentos correspondientes.

* Datos que el sistema necesita:

Monto de la compra (número real).

Condición de cliente VIP (True/False).

Condición de código de descuento especial (True/False).

* Proceso:

Si la compra supera los $100, aplica el 20% de descuento.

Si el cliente es VIP, aplicar 10% de descuento adicional.

Si el cliente tiene código especial, aplica el 5% de descuento adicional.

Calcular el total restando los descuentos al monto inicial.

* Resultado:

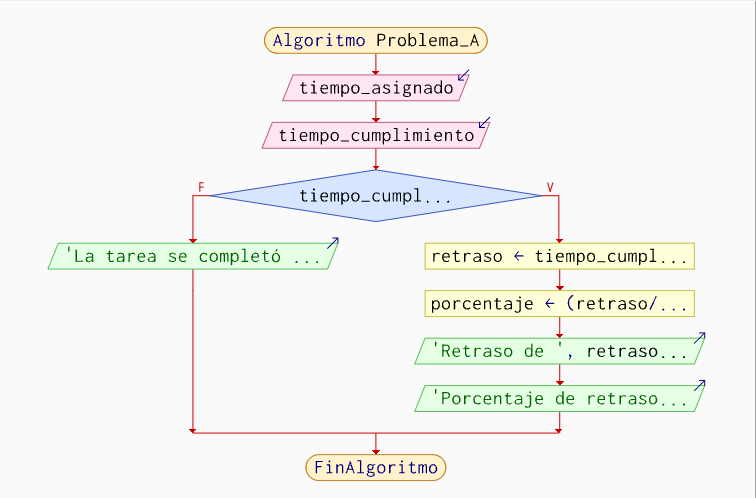
Mostrar el total a pagar luego de aplicar los descuentos.

1. HERRAMIENTAS

* PSEINT (Diagrama de flujo)
* PYTHON
* GIT
* GITHUB
* VS CODE

1. PSEUDOCODIGO DE CADA UNO DE LOS PROBLEMAS

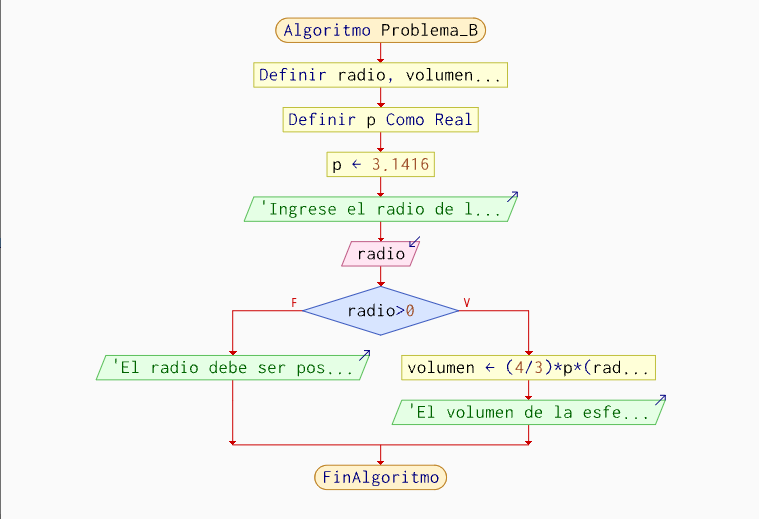
## Algoritmo A:



*Figura 2 pseudocódigo del algoritmo (A), fuente propia*

En la figura 2, empieza la ejecución con el nombre Problema\_A. Luego solicita dos datos de entrada que son : Tiempo\_asignado (el tiempo en el que debería cumplir la tarea) y tiempo\_cumplimiento (tiempo en el que cumple la tarea), luego condicionamos que si el tiempo\_cumplimiento es mayor al tiempo\_asignado, se calcula el porcentaje de retraso que tuvo dependiendo de la cantidad de días de atraso que tuvo por el 100% de cumpliento y así calculamos el porcentaje de efectividad al cumplir la tarea. Y si esto no se cumple, significa que la tarea se completó dentro del tiempo estipulado.

## Algoritmo B:



## Figura 3 pseudocódigo del algoritmo B, Fuente propia.

En la figura 3, se ejecuta con el nombre Problema\_B, luego definimos el radio, volumen y P como reales (lo que significa que pueden tener decimales), luego en P añadimos el valor de número *pi(3.1416 )*y pedimos al usuario que nos de el radio de la esfera validando que este sea positivo, si el radio es negativo solicita un valor estrictamente positivo y cuando cumpla que el radio es positivo el volumen será igual a *V=4/3\*pi\*(radio solicitado)^3* y con esta fórmula hallamos el valor del volumen de la esfera para su producción.

## Algoritmo C:

## 

*Figura 4 pseudocódigo del algoritmo (C),fuente propia.*

En la figura 4, la ejecución comienza con el nombre Problema\_C. A continuación, se solicitan tres datos de entrada: nivel\_requerido (nivel mínimo de acceso necesario), tarjeta\_activa (estado de la tarjeta de acceso) y días\_contrasena (tiempo transcurrido desde el último cambio de contraseña). Posteriormente, se establece una condición que valida tres aspectos al mismo tiempo: que el nivel\_usuario sea mayor o igual al nivel\_requerido, que la tarjeta\_activa sea verdadera y que los días\_contrasena sean menores o iguales a 30. Si las tres condiciones se cumplen, el sistema muestra el mensaje “Acceso permitido”; en caso contrario, se muestra el mensaje “Acceso denegado”.

## Algoritmo D:

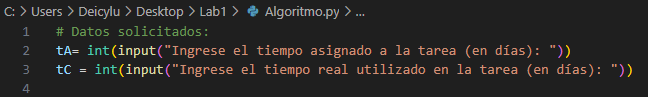
## 

*Figura 5: pseudocódigo del algoritmo D, fuente propia.*

En la figura 5, la ejecución comienza con el nombre Problema D. Primero se leen tres datos de entrada: monto\_compra (valor de la compra realizada), es\_vip (si el cliente pertenece al programa VIP) y tiene\_codigo (si el cliente tiene un código de descuento). Posteriormente, el algoritmo asigna a la variable total el valor inicial del monto\_compra, adicional tiene tres condiciones: .Si el monto\_compra es mayor a 100, se descuenta un 20% del total,Si el cliente es VIP, se descuenta un 10% adicional y Si el cliente tiene código de descuento, se descuenta un 5% adicional. Finalmente, después de aplicar los descuentos que correspondan, el algoritmo muestra en pantalla el mensaje con el total a pagar.

1. CODIFICACIÓN

## **Algoritmo A:**



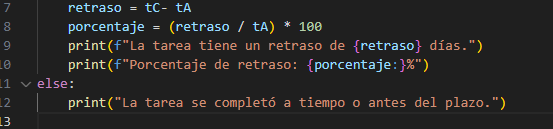
*Figura 6: código del primer algoritmo (A), fuente propia.*

En la figura 6 pide al usuario que ingrese el tiempo planificado para completar la tarea, en días, y lo guarda como un número entero en la variable tA.(tiempo Asignado) Y pide al usuario ingresar el tiempo real que se utilizó para completar la tarea, también en días y como número entero, guardándolo en tC.



*Figura 7 código del primer algoritmo (A), fuente propia.*

Compara si el tiempo real usado (tC) es mayor que el tiempo asignado (tA). Es decir, si la tarea se terminó después del tiempo planeado.



*Figura 8: código del primer algoritmo (A), fuente propia.*

En la figura 8: si en la figura 7 se cumple la condición, se ejecuta que el retraso = tC - tA calcula cuántos días de retraso hubo. Luego calcula el porcentaje = (retraso / tA) \* 100. Calcula qué porcentaje representa ese retraso respecto al tiempo asignado.Si la tarea se completó a tiempo o antes (cuando tC no es mayor que tA):Se imprime: "La tarea se completó a tiempo o antes del plazo."

## **Algoritmo B:**

## 

*Figura 9: código del primer algoritmo (B), fuente propia.*

En la Figura 9 se importa la librería de math que se usará más adelante en el código.



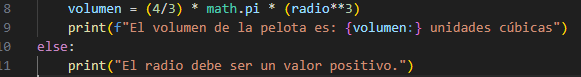
*Figura 10: código del primer algoritmo (B), fuente propia.*

En la figura 10: solicita al usuario que ingrese el radio de la pelota. Se convierte a un número decimal (float) para permitir valores con decimales.



*Figura 11: código del primer algoritmo (B), fuente propia.*

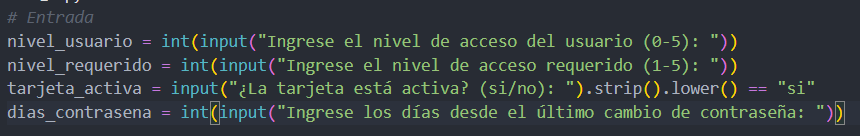
En la figura 11 verifica que el radio ingresado sea un número positivo, porque el radio debe ser mayor que cero para que tenga sentido calcular el volumen.



*Figura 12: código del primer algoritmo (B), fuente propia.*

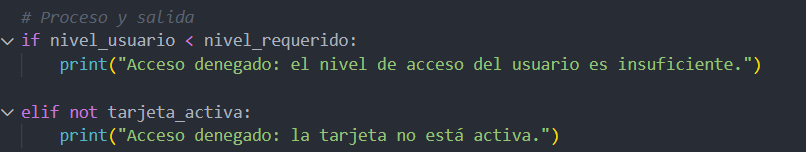
En la figura 12:calcula el volumen de la esfera con la fórmula (4/3\*pi\*radio^3) donde math.pi es el valor de π y radio^3 es el radio elevado al cubo.Muestra el resultado del volumen con dos decimales.Si el radio no es positivo (cero o negativo):Imprime el mensaje: "El radio debe ser un valor positivo."

**Algoritmo C:**



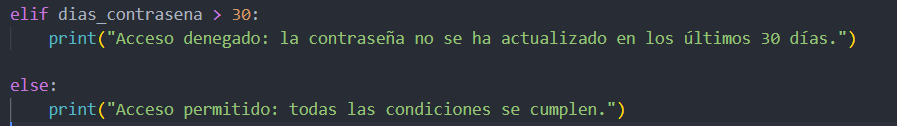
*Figura 13: código del primer algoritmo (C), fuente propia.*

En la figura 13 pide al usuario que ingrese el acceso del usuario de (0-5), se pide el acceso mínimo requerido que necesita para entrar de (1-5), se le pregunta al usuario si la tarjeta está activa o no está activa y se le pide que ingrese los días desde el último cambio de contraseña que realizó.



*Figura 14: código del primer algoritmo (C), fuente propia*

En la figura 14: se compara el nivel de acceso que tiene el usuario (nivel\_usuario) con el nivel mínimo necesario (nivel\_requerido) y verifica si la tarjeta no está activa con el operador lógico not haciendo que tarjeta activa-true y tarjeta inactiva-false y el operador not invierte el valor, si la tarjeta activa es False, not tarjeta activa será true.



*Figura 15: código del primer algoritmo (C), fuente propia*

*En la figura 15: se verifica si el número de días desde el último cambio de contraseña es mayor que 30. Si la condicion es verdadera le dara el mensaje acceso denegado:la contraseña no se ha actualizado en los ultimos 30 dias y si la condicion es falsa le dara el mensaje de acceso permitido: Todas las condiciones se cumplen*

## **Algoritmo D:**



*Figura 16: código del primer algoritmo (D), fuente propia.*

En la figura 16 pide al usuario ingresar el monto total de la compra, como número decimal.



*Figura 17: código del primer algoritmo (D), fuente propia.*

En la figura 17 pregunta si el cliente es VIP; guarda True si la respuesta es "si" y False en caso contrario*.*



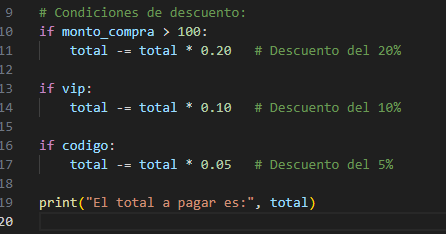
*Figura 18: código del primer algoritmo (D), fuente propia.*

En la figura 18: Pregunta si tiene un código de descuento; guarda True si la respuesta es "sí", False si no.



*Figura 19: código del primer algoritmo (D), fuente propia.*

En la figura 19: Inicializa la variable total con el monto original de la compra.

**

*Figura 20: código del primer algoritmo (D), fuente propia.*

En la figura 20 Si la compra es mayor a 100, aplica un 20% de descuento sobre el total actual. Si el cliente es VIP, aplica un 10% de descuento sobre el total resultante después del descuento anterior. Imprime el total a pagar con todos los descuentos aplicados uno tras otro.

III. EVALUACIÓN Y PRUEBAS

Tablas de pruebas Algoritmo A:

|  |  |
| --- | --- |
| Código | Descripción de la prueba |
|  | En esta evidencia podemos ver que cuando el tiempo asignado es mayor al de tiempo completo o hay necesidad de dar el retraso. |
|  | Si el número de días asignados es negativo lo toma y realiza la operación se tendrá en cuenta para futuros cambios |
|  | Si el número de días para completar la tarea es negativo se ejecuta como si cumpliera el tiempo |
|  | Si en las dos variables es negativo se ejecuta como si estuviera dentro del lapso |
|  | Si el tiempo de completar la tarea es mayor a el del tiempo asignado se da el tiempo del retraso y el porcentaje del mismo. |

Tabla de pruebas algoritmo B:

|  |  |
| --- | --- |
| Código | Descripción de la prueba |
|  | Si el radio en un número entero se ejecuta normalmente y da el volumen de la pelota. |
|  | Si el número es negativo, el código solicita que se ingrese un número positivo. |
|  | Si el número es decimal, el código se ejecuta normalmente. |
|  | Si el número es decimal negativo el código solicita que sea un número positivo |
|  | Si el valor es 0, solicita que el valor sea mayor a este. |

Tabla de pruebas Algoritmo C:

|  |  |
| --- | --- |
| Código | Descripción de la prueba |
|  | Prueba realizada con negativos en nivel usuario y requerido con tarjeta activa y 23 días desde el último cambio dando acceso. |
|  | Prueba realizada con el número 0 en ambos casos, con tarjeta activa y el último cambio con 25 días dando acceso. |
|  | Prueba realizada con el nivel de usuario 1 y el requerido 0 y tarjeta inactiva con 30 días desde el último cambio de contraseña denegando acceso. |
|  | Prueba realizada con el nivel de usuario 1 y el requerido en 4 y tarjeta activa con 28 días desde el último cambio de contraseña denegando acceso. |
|  | Prueba realizada con el nivel de usuario 1 y el requerido 4 pero con tarjeta activa y días mayor a 35 los días de el último cambio denegando acceso. |

Tabla de pruebas Algoritmo D :

|  |  |
| --- | --- |
| Código | Descripción de la prueba |
|  | Si la compra es menor a 100 no aplica el descuento y si no es VIP y no tiene código de descuento el valor a pagar es el mismo. |
|  | Si la compra es menor a 100 no aplica el descuento pero si es VIP y no tiene código de descuento, solo aplica el porcentaje de descuento del VIP |
|  | Si la compra es menor a 100 no aplica el descuento y si no es VIP y si tiene código de descuento, solo aplica el porcentaje de descuento del código de descuento. |
|  | Si la compra es menor a 100 no aplica el descuento y si tiene VIP y si tiene código de descuento, aplica el porcentaje de descuento del código de descuento y del VIP. |
|  | Si la compra es mayor a 100, aplica el descuento y si tiene VIP y si tiene código de descuento, aplica el porcentaje de descuento del código de descuento y del VIP. |

III. CONCLUSIONES

1. Los algoritmos permiten estructurar y resolver problemas reales de manera lógica y ordenada, facilitando su implementación en programas computacionales.
2. La aplicación de la programación en distintos contextos (proyectos de software, manufactura, seguridad y comercio electrónico) evidencia su relevancia en la optimización de procesos y en la toma de decisiones.
3. El uso de herramientas como Python contribuye al aprendizaje práctico, ya que convierte los conceptos teóricos en soluciones funcionales y comprensibles.
4. El análisis y diseño de algoritmos fortalecen competencias esenciales en ingeniería de sistemas, como el pensamiento lógico, la abstracción y la capacidad para enfrentar problemas en diferentes áreas profesionales.
5. La resolución de problemas mediante algoritmos fomenta la innovación y la adaptación, ya que permite generar soluciones aplicables a diferentes escenarios y ajustables a los cambios en los requerimientos o condiciones del entorno.

IV. REFERENCIAS

Araújo, J. (2015). Una solución de servicio integrada para el procesamiento de cupones de descuento digitales. *Springer Nature*, *353*(1), 395-401. <https://link.springer.com/chapter/10.1007/978-3-319-16486-1_39?utm_source>

labex. (2017, N/A N/A). *calcular volumen*. Determinar el volumen de una esfera en C. <https://labex.io/es/tutorials/c-determine-the-volume-of-a-sphere-in-c-435173?utm_source>

Shieh, S. P., & Yang, W. H. (1999). Esquemas de autenticación de contraseñas con tarjetas inteligentes. *ELSEVIER*, *18*(8), 727-733. <https://www.sciencedirect.com/science/article/abs/pii/S0167404899801369?utm_source>

Javed, R. (n.d.). *calculo del porcentaje de retraso*. Calculadora de porcentaje de retraso. Retrieved 09 24, 2025, from <https://calculatorshub.net/es/tools/delay-percentage-calculator/?utm_source>