

POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> El programa solicita al usuario ingresar un número natural. Calcula la suma de los números naturales desde 1 hasta el número ingresado. 	<ul style="list-style-type: none"> No se valida si el número ingresado es realmente un número natural (mayor o igual a 1). No se manejan errores si el usuario ingresa un valor no numérico. 	<ul style="list-style-type: none"> El uso de bucles y acumuladores para calcular la suma. Podríamos mejorar el programa agregando validaciones para asegurarnos de que el usuario ingrese un número natural válido.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> El programa recorre un bucle desde 100 hasta 1. Si el número es impar, lo muestra en la salida. 	<ul style="list-style-type: none"> El código podría beneficiarse de comentarios explicativos para facilitar la comprensión. 	<ul style="list-style-type: none"> El uso del operador % para verificar si un número es impar. La elección de comenzar desde 100 y contar hacia atrás.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> El programa recorre un bucle desde 1 hasta 100. Acumula la suma de los cuadrados en la variable sum. 	<ul style="list-style-type: none"> No se manejan errores si ocurre algún problema durante la ejecución. 	<ul style="list-style-type: none"> El uso de la fórmula i^2 para calcular el cuadrado de cada número. La elección de recorrer los números desde 1 hasta 100.

POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> El programa solicita al usuario ingresar las notas de 10 estudiantes. Calcula la suma de todas las notas ingresadas. 	<ul style="list-style-type: none"> No se proporciona ninguna descripción o mensaje al usuario sobre lo que está sucediendo en el programa. 	<ul style="list-style-type: none"> La elección de recorrer 10 estudiantes y calcular el promedio general es relevante para un contexto escolar.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> Determina cuántos estudiantes aprobaron (notas mayores o iguales a 70) y cuántos reprobaron. Muestra la cantidad de alumnos aprobados, reprobados y el promedio general del grupo. 	<ul style="list-style-type: none"> No se valida si las notas ingresadas están dentro del rango válido (por ejemplo, entre 0 y 100). 	<ul style="list-style-type: none"> El uso de un arreglo para almacenar las notas de los estudiantes. La lógica para determinar si un estudiante aprobó o reprobó según la calificación.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> Verifica si cada número es par (usando el operador %). Si el número es par, lo suma al acumulador sum. 	<ul style="list-style-type: none"> No se proporciona ninguna descripción o mensaje al usuario sobre lo que está sucediendo en el programa. No se manejan errores si ocurre algún problema durante la ejecución. 	<ul style="list-style-type: none"> El uso del operador % para verificar si un número es par. La elección de recorrer los números desde 100 hasta 200.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> El programa permite al usuario ingresar la longitud de los vectores y los elementos de los vectores. Calcula la suma de los vectores elemento a elemento. 	<ul style="list-style-type: none"> No se manejan errores si los vectores no tienen la misma longitud. 	<ul style="list-style-type: none"> El uso de vectores para almacenar los elementos. La lógica para calcular la suma de los vectores es relevante para aplicaciones matemáticas y científicas.

POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> • El código define dos vectores, vect_A y vect_B, y calcula su producto punto correctamente. • Utiliza bucles for para ingresar los elementos de los vectores desde la entrada estándar. 	<ul style="list-style-type: none"> • No hay manejo de errores para verificar si los valores ingresados son números válidos. • Los vectores están fijos en tamaño (3 elementos). Sería más flexible si el usuario pudiera especificar el tamaño. 	<ul style="list-style-type: none"> • El cálculo del producto punto es una operación matemática fundamental en álgebra lineal. • El uso de bucles for para iterar a través de los elementos de los vectores es una buena práctica.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> • El código solicita al usuario las dimensiones de dos matrices y verifica si son compatibles para la multiplicación. • Utiliza bucles anidados para ingresar los elementos de las matrices desde la entrada estándar. 	<ul style="list-style-type: none"> • No hay manejo de errores para verificar si las dimensiones ingresadas son números válidos o si las matrices son cuadradas. • Las matrices están fijas en tamaño según las dimensiones ingresadas por el usuario. Sería más flexible permitir matrices de tamaño variable. 	<ul style="list-style-type: none"> • La multiplicación de matrices es una operación fundamental en álgebra lineal y tiene aplicaciones en gráficos por computadora, procesamiento de imágenes y más. • El uso de bucles anidados para recorrer los elementos de las matrices es útil.
POSITIVO	NEGATIVO	INTERESANTE
<ul style="list-style-type: none"> • Calcula correctamente la matriz transpuesta intercambiando filas por columnas. 	<ul style="list-style-type: none"> • El código carece de comentarios que expliquen su funcionamiento. 	<ul style="list-style-type: none"> • La matriz transpuesta es útil en álgebra lineal y tiene aplicaciones en transformaciones geométricas y programación gráfica.