

Yo	BlackBox
<pre> /*Dado un número natural n se desea calcular la suma de los números naturales desde 1 hasta n. Codifica el programa que resuelva este planteamiento.*/ #include <iostream> using namespace std; void sumatoria(int num, int &suma); main() { int num, suma = 0; cout << "Digite un numero natural hasta el que desee sumar: "; cin >> num; sumatoria(num, suma); printf("La sumatoria de todos los numeros es %i", suma); return 0; } void sumatoria(int num, int &suma) { suma = 0; for (int i = 1; i <= num; i++) { suma += i; } } </pre>	<pre> /*Dado un número natural n se desea calcular la suma de los números naturales desde 1 hasta n. Codifica el programa que resuelva este planteamiento.*/ #include <iostream> using namespace std; int sumatoria(int num); int main() { int num; cout << "Digite un numero natural hasta el que desee sumar: "; cin >> num; int suma = sumatoria(num); cout << "La sumatoria de todos los numeros es " << suma << endl; return 0; } int sumatoria(int num) { int suma = 0; for (int i = 1; i <= num; i++) { suma += i; } return suma; } </pre>

Constructivismo - Ejercicio #1		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> La IA implementó una función con parámetros. Usó solo cout y no printf. La IA no tuvo que usar "&" . 	<ul style="list-style-type: none"> Con el printf es más sencillo, la IA usó cout. 	<ul style="list-style-type: none"> La IA define con facilidad los valores en medio del proceso y no antes.

Yo	BlackBox
----	----------

<pre> /*Desarrollar un programa que imprima los números impares en orden descendente que hay entre 1 y 100.*/ #include <iostream> using namespace std; void mostrarImpares(); main() { mostrarImpares(); return 0; } void mostrarImpares() { for (int i = 100; i >= 1; i--) { if (i % 2 != 0) { printf("%i\n", i); } } } </pre>	<pre> /*Desarrollar un programa que imprima los números impares en orden descendente que hay entre 1 y 100.*/ #include <iostream> using namespace std; void mostrarImpares(); int main() { mostrarImpares(); return 0; } void mostrarImpares() { for (int i = 100; i >= 1; i--) { if (i % 2 != 0) { cout << i << endl; } } } </pre>
---	---

Constructivismo - Ejercicio #2		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> Usó cout envés de printf 	Nada en particular.	<ul style="list-style-type: none"> Ahora ambos códigos son casi iguales.

Yo	BlackBox
<pre> /*Escriba el programa para calcular la suma de los cuadrados de los números entre 1 y 100.*/ #include <iostream> using namespace std; void sumatoria(); main() { sumatoria(); return 0; } void sumatoria(){ int suma; for (int i = 1; i <= 100; i++) { suma += i*i; } printf("La sumatoria de los cuadrados de los numeros entre 1 y 100 es %i", suma); } </pre>	<pre> /*Escriba el programa para calcular la suma de los cuadrados de los números entre 1 y 100.*/ #include <iostream> using namespace std; int main() { int sum = 0; for (int i = 1; i <= 100; i++) { sum += i * i; } cout << "The sum of the squares of the numbers between 1 and 100 is " << sum << endl; return 0; } </pre>

Constructivismo - Ejercicio #3		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> Ahora es mi código el que es un poco más “complejo”. La IA usó cout. 	Nada en particular.	<ul style="list-style-type: none"> BlackBox puso en inglés el código. Le agrega “int” al main.

Yo	BlackBox
<pre> /*Realice un programa que permita calcular y dar como salida el promedio general de una sección, tomando en cuenta que está compuesta por 10 estudiantes y que se tiene la nota de cada uno de ellos.*/ #include <iostream> using namespace std; void calcularPromedio(); main() { calcularPromedio(); return 0; } void calcularPromedio(){ float nota; float suma=0; float promedio; for (float i = 1; i <= 10; i++) { cout << "Digite la nota del estudiante: "; cin >> nota; suma += nota; } promedio=suma/10; printf("El promedio de los 10 estudiantes es %.2f", promedio); } </pre>	<pre> /*Realice un programa que permita calcular y dar como salida el promedio general de una sección, tomando en cuenta que está compuesta por 10 estudiantes y que se tiene la nota de cada uno de ellos.*/ #include <iostream> #include <iomanip> using namespace std; float calcularPromedio(); int main() { float promedio = calcularPromedio(); cout << "El promedio de los 10 estudiantes es " << fixed << setprecision(2) << promedio << endl; return 0; } float calcularPromedio(){ float nota; float suma=0; for (int i = 1; i <= 10; i++) { cout << "Digite la nota del estudiante " << i << ": "; cin >> nota; suma += nota; } return suma/10; } </pre>

Constructivismo - Ejercicio #4		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> La IA hizo uso de una función que no era void, pero tampoco tenía parámetros. 	<ul style="list-style-type: none"> La IA alargó innecesariamente el código. 	<ul style="list-style-type: none"> La IA hizo uso de una biblioteca nueva. Y también uso de operadores como "fixed" o "setprecision".

Yo	BlackBox
----	----------

<pre> /*Para un grupo de clase de 8 estudiantes, realice un programa que permita calcular y dar como salida lo siguiente: ✓ Cantidad de alumnos aprobados ✓ Cantidad de alumnos reprobados ✓ Promedio general del grupo*/ #include <iostream> using namespace std; void calculo(); main() { calculo(); return 0; } void calculo(){ float nota; float suma=0; float promedio; int aprobado=0; int reprobado=0; for (float i = 1; i <= 8; i++) { cout << "Digite la nota del estudiante: "; cin >> nota; suma += nota; if (nota >= 70) { aprobado += 1; } else { reprobado += 1; } } promedio=suma/8; printf("La cantidad de alumnos aprobados es de %.i\n", aprobado); printf("La cantidad de alumnos reprobados es de %.i\n", reprobado); printf("El promedio de los 10 estudiantes es %.2f\n", promedio); </pre>	<pre> /*Para un grupo de clase de 8 estudiantes, realice un programa que permita calcular y dar como salida lo siguiente: ✓ Cantidad de alumnos aprobados ✓ Cantidad de alumnos reprobados ✓ Promedio general del grupo*/ #include <iostream> #include <iomanip> using namespace std; void calcularPromedio(float notas[], int n, int& aprobados, int& reprobados, float& promedio); int main() { const int n = 10; float notas[n]; int aprobados, reprobados; float promedio; for (int i = 0; i < n; i++) { cout << "Digite la nota del estudiante " << i+1 << ": "; cin >> notas[i]; } calcularPromedio(notas, n, aprobados, reprobados, promedio); cout << "La cantidad de alumnos aprobados es de " << aprobados << endl; cout << "La cantidad de alumnos reprobados es de " << reprobados << endl; cout << "El promedio de los 10 estudiantes es " << fixed << setprecision(2) << promedio << endl; return 0; } void calcularPromedio(float notas[], int n, int& aprobados, int& reprobados, float& promedio){ float suma=0; for (int i = 0; i < n; i++) { suma += notas[i]; if (notas[i] >= 70) { aprobados += 1; } else { reprobados += 1; } } promedio=suma/n; } </pre>
--	---

Constructivismo - Ejercicio #5		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> La IA implementó una función con parámetros. También hizo uso de un arreglo para las notas. 	<ul style="list-style-type: none"> La IA alargó innecesariamente el código. 	<ul style="list-style-type: none"> La IA hizo uso de una biblioteca nueva.

Yo	BlackBox
----	----------

<pre> /*Hacer un programa que sume los números pares comprendidos entre 100 y 200.*/ #include <iostream> using namespace std; void sumaPares(); main() { sumaPares(); return 0; } void sumaPares(){ int suma=0; for (int i = 100; i <= 200; i++) { if (i % 2 == 0) { suma+=i; } } printf("La suma de los numeros pares entre 100 y 200 es %i", suma); } </pre>	<pre> /*Hacer un programa que sume los números pares comprendidos entre 100 y 200.*/ #include <iostream> using namespace std; int main() { int suma = 0; for (int i = 100; i <= 200; i++) { if (i % 2 == 0) { suma += i; } } cout << "La suma de los numeros pares entre 100 y 200 es " << suma << endl; return 0; } </pre>
--	---

Constructivismo - Ejercicio #6		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> La IA simplificó el código. Hizo uso de cout. 	Nada en particular	Nada en particular.

Yo	BlackBox
<pre> /*Desarrolla un programa que solicite al usuario ingresar dos vectores de igual longitud y luego calcule la suma de ambos vectores. El resultado debe mostrarse en un tercer vector.*/ #include <iostream> using namespace std; void sumaVectores(); main() { sumaVectores(); return 0; } void sumaVectores(){ int arreglo1[3]; int arreglo2[3]; int resultado[3]; cout << "Ingrese los 3 elementos del primer arreglo\n"; for (int i = 0; i < 3; i++) { cout << "Ingrese elemento: "; cin >> arreglo1[i]; } cout << "Ingrese los 3 elementos del segundo arreglo\n"; for (int i = 0; i < 3; i++) { </pre>	<pre> /*Desarrolla un programa que solicite al usuario ingresar dos vectores de igual longitud y luego calcule la suma de ambos vectores. El resultado debe mostrarse en un tercer vector.*/ #include <iostream> using namespace std; void sumaVectores(); int main() { sumaVectores(); return 0; } void sumaVectores() { int arreglo1[3]; int arreglo2[3]; int resultado[3]; cout << "Ingrese los 3 elementos del primer arreglo\n"; for (int i = 0; i < 3; i++) { cout << "Ingrese elemento: "; cin >> arreglo1[i]; } cout << "Ingrese los 3 elementos del segundo arreglo\n"; for (int i = 0; i < 3; i++) { cout << "Ingrese elemento: "; cin >> arreglo2[i]; } cout << "La suma de los dos arreglos es:\n"; for (int i = 0; i < 3; ++i) { cout << resultado[i] << " "; } } </pre>

<pre> cout << "Ingrese elemento: "; cin >> arreglo2[i]; } for (int i = 0; i < 3; ++i) { resultado[i] = arreglo1[i] + arreglo2[i]; } cout << "La suma de los dos arreglos es:" << endl; for (int i = 0; i < 3; ++i) { cout << resultado[i] << " "; } } </pre>	
--	--

Constructivismo - Ejercicio #7		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> Los códigos son casi iguales, así que se puede asumir que lo hice bien. 	Nada en particular.	Nada en particular.

Yo	BlackBox
<pre> /*Crea un programa que permita al usuario ingresar dos vectores de la misma longitud y luego calcule su producto punto. Muestra el resultado al finalizar el cálculo.*/ #include <iostream> using namespace std; void multiVectores(); main() { multiVectores(); return 0; } void multiVectores(){ int arreglo1[3]; int arreglo2[3]; int resultado[3]; cout << "Ingrese los 3 elementos del primer arreglo\n"; for (int i = 0; i < 3; i++) { cout << "Ingrese elemento: "; cin >> arreglo1[i]; } cout << "Ingrese los 3 elementos del segundo arreglo\n"; for (int i = 0; i < 3; i++) { </pre>	<pre> /*Crea un programa que permita al usuario ingresar dos vectores de la misma longitud y luego calcule su producto punto. Muestra el resultado al finalizar el cálculo.*/ #include <iostream> using namespace std; void multiVectores(int arr1[], int arr2[], int n); int main() { int arr1[3] = {0}; int arr2[3] = {0}; int n = sizeof(arr1)/sizeof(arr1[0]); cout << "Enter the elements of the first vector: "; for (int i = 0; i < n; i++) cin >> arr1[i]; cout << "Enter the elements of the second vector: "; for (int i = 0; i < n; i++) cin >> arr2[i]; multiVectores(arr1, arr2, n); return 0; } void multiVectores(int arr1[], int arr2[], int n) { int result = 0; for (int i = 0; i < n; i++) result += arr1[i] * arr2[i]; cout << "The dot product of the two vectors is: " << result << endl; } </pre>

<pre> cout << "Ingrese elemento: "; cin >> arreglo2[i]; } for (int i = 0; i < 3; ++i) { resultado[i] = arreglo1[i] * arreglo2[i]; } cout << "La suma de los dos arreglos es:" << endl; for (int i = 0; i < 3; ++i) { cout << resultado[i] << " "; } } </pre>	
--	--

Constructivismo - Ejercicio #8		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> La IA hizo uso de una función con parámetros. 	Nada en particular.	<ul style="list-style-type: none"> Hizo unas extrañas equivalencias, como en la definición de los arreglos. Hizo uso del operador "sizeof". Escribió el código en inglés.

Yo	BlackBox
<pre> /*Diseña un programa que solicite al usuario ingresar dos matrices y luego realice la multiplicación de matrices. Asegúrate de que las dimensiones de las matrices sean compatibles para la multiplicación y muestra la matriz resultante.*/ #include <iostream> using namespace std; void multiMatrices(); int main() { multiMatrices(); return 0; } void multiMatrices(){ int matriz1[2][2]; int matriz2[2][2]; int matriz3[2][2]; cout << "Ingrese los elementos de la primera matriz:" << endl; for (int i = 0; i < 2; ++i) /*fila*/ { for (int j = 0; j < 2; ++j) /*columna*/ { cout << "Ingrese el elemento (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz1[i][j]; } } } </pre>	<pre> /*Diseña un programa que solicite al usuario ingresar dos matrices y luego realice la multiplicación de matrices. Asegúrate de que las dimensiones de las matrices sean compatibles para la multiplicación y muestra la matriz resultante.*/ #include <iostream> using namespace std; void multiMatrices(); int main() { multiMatrices(); return 0; } void multiMatrices(){ int matriz1[2][2]; int matriz2[2][2]; int matriz3[2][2]; cout << "Ingrese los elementos de la primera matriz:" << endl; for (int i = 0; i < 2; ++i) /*fila*/ { for (int j = 0; j < 2; ++j) /*columna*/ { cout << "Ingrese el elemento (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz1[i][j]; } } } </pre>

<pre> cout << "Ingrese los elementos de la segunda matriz:" << endl; for (int i = 0; i < 2; ++i) /*fila*/ { for (int j = 0; j < 2; ++j) /*columna*/ { cout << "Ingrese el elemento (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz2[i][j]; } } for (int i = 0; i < 2; ++i) { for (int j = 0; j < 2; ++j) { matriz3[i][j] = matriz1[i][j] * matriz2[i][j]; } } cout << "La matriz resultante es:" << endl; for (int i = 0; i < 2; ++i) { for (int j = 0; j < 2; ++j) { cout << matriz3[i][j] << " "; } cout << endl; } } </pre>	<pre> cout << "Ingrese los elementos de la segunda matriz:" << endl; for (int i = 0; i < 2; ++i) /*fila*/ { for (int j = 0; j < 2; ++j) /*columna*/ { cout << "Ingrese el elemento (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz2[i][j]; } } for (int i = 0; i < 2; ++i) { for (int j = 0; j < 2; ++j) { matriz3[i][j] = 0; // Initialize matriz3 elements to 0 for (int k = 0; k < 2; ++k) { matriz3[i][j] += matriz1[i][k] * matriz2[k][j]; } } } cout << "La matriz resultante es:" << endl; for (int i = 0; i < 2; ++i) { for (int j = 0; j < 2; ++j) { cout << matriz3[i][j] << " "; } cout << endl; } } </pre>
---	--

Constructivismo - Ejercicio #9		
Positivo	Negativo	Interesante
<ul style="list-style-type: none"> Los códigos son casi iguales, así que se puede asumir que lo hice bien. 	Nada en particular.	Nada en particular.

Yo	BlackBox
<pre> /*Implementa un programa que solicite al usuario ingresar una matriz y luego calcule su transposición. La transposición de una matriz se realiza intercambiando filas por columnas. Imprime la matriz transpuesta como salida.*/ #include <iostream> using namespace std; void transposicion(); main() { transposicion(); return 0; } void transposicion(){ int matriz1[3][3]; int matriz2[3][3]; cout << "Ingrese los elementos de la matriz:" << endl; for (int i = 0; i < 3; ++i) /*fila*/ { for (int j = 0; j < 3; ++j) /*columna*/ { cout << "Ingrese el elemento (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz1[i][j]; } } } </pre>	<pre> /*Implementa un programa que solicite al usuario ingresar una matriz y luego calcule su transposición. La transposición de una matriz se realiza intercambiando filas por columnas. Imprime la matriz transpuesta como salida.*/ #include <iostream> using namespace std; void transposeMatrix(); int main() { transposeMatrix(); return 0; } void transposeMatrix(){ int matriz1[3][3]; int matriz2[3][3]; cout << "Enter the elements of the matrix:" << endl; for (int i = 0; i < 3; ++i) // row { for (int j = 0; j < 3; ++j) // column { cout << "Enter the element (" << i + 1 << ", " << j + 1 << "): "; cin >> matriz1[i][j]; } } } </pre>


```

for (int i = 0; i < 3; ++i)
{
    for (int j = 0; j < 3; ++j)
    {
        matriz2[j][i] = matriz1[i][j];
    }
}

cout << "La matriz transpuesta es:" << endl;
for (int i = 0; i < 3; ++i)
{
    for (int j = 0; j < 3; ++j)
    {
        cout << matriz2[i][j] << " ";
    }
    cout << endl;
}
}

```

```

for (int i = 0; i < 3; ++i)
{
    for (int j = 0; j < 3; ++j)
    {
        matrix2[j][i] = matrix1[i][j];
    }
}

cout << "The transposed matrix is:" << endl;
for (int i = 0; i < 3; ++i)
{
    for (int j = 0; j < 3; ++j)
    {
        cout << matrix2[i][j] << " ";
    }
    cout << endl;
}
}

```

Constructivismo - Ejercicio #10

Positivo	Negativo	Interesante
<ul style="list-style-type: none"> Los códigos son casi iguales, así que se puede asumir que lo hice bien. 	Nada en particular.	<ul style="list-style-type: none"> Escribió el código en inglés, hasta las variables.