

## **Lista de Ejercicios**

### **Vectores y Matrices**

1. Implementar un algoritmo en C++ que lea el tamaño de un vector y luego los elementos de este vector. Después de eso, el algoritmo debe llamar a función que calcule la suma de todos los elementos del vector.
2. Implementar un algoritmo en C++ que lea el tamaño de un vector y luego los elementos de este vector. Después de eso, el algoritmo debe llamar a función que calcule la diferencia entre la suma de elementos en posiciones impares y la suma de los elementos en posiciones pares.
3. Se tienen las frecuencias absolutas de “n” elementos. Implementar un algoritmo en C++ que calcule la frecuencia relativa y la frecuencia relativa acumulada de los elementos e imprima ambos valores junto con las frecuencias absolutas.
4. Implementar un algoritmo en C++ que lea el tamaño de un vector y luego los elementos de este vector. Los elementos del vector deben estar ordenados de manera ascendente. Implementar el algoritmo de búsqueda binaria que retorne si el elementos se encuentra en el vector o no.
5. Implementar un algoritmo en C++ que realice el producto interno de dos vectores.
6. Implementar un algoritmo en C++ que reciba un vector de enteros y un número entero que desea ser buscado. El algoritmo debe retornar el número de veces que el número aparece en el vector.
7. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz para después calcular la suma de todos los elementos de la matriz.
8. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz para después calcular la suma de los elementos por cada fila.
9. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz para después calcular la suma de los elementos por cada columna.
10. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz para después calcular el número máximo de una fila específica.
11. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz para después calcular la suma de los elementos de la diagonal principal.
12. Implementar un algoritmo en C++ que lea el tamaño de una matriz bidimensional y luego los elementos de la matriz e imprima la transpuesta de la matriz.
13. Implementar un algoritmo en C++ que reciba dos matrices y realice la suma de las mismas.
14. Implementar un algoritmo en C++ que reciba dos matrices y que realice la multiplicación de las mismas.
15. Implementar un algoritmo en C++ que reciba una matriz y determine si la matriz es triangular superior o no.
16. Implementar un algoritmo en C++ que reciba una matriz y luego pueda realizar las dos siguientes operaciones en matrices: intercambiar dos filas, sumar la primera fila a la segunda y atribuirla a la segunda.