## Factorial de un número (Recursividad)

Calcula la factorial de un número dado de forma recursiva. El algoritmo multiplica el número actual por la factorial del número anterior hasta llegar a 1.

## Suma de una lista de números (Recursividad)

Implementa una función recursiva que suma todos los elementos de una lista. El algoritmo toma el último elemento de la lista y lo suma a la suma de los elementos restantes.

## Serie de Fibonacci (Recursividad)

Obtiene el enésimo número de la serie de Fibonacci de manera recursiva. El algoritmo suma los dos números anteriores de la serie hasta alcanzar el índice deseado.

## Búsqueda binaria (Recursiva)

Realiza una búsqueda binaria en una lista ordenada usando recursión. El algoritmo divide la lista en mitades y busca el elemento en la mitad, repitiendo el proceso con la sublista correspondiente.

## Potencia de un número (Recursividad)

Calcula la potencia de un número `a^b` de forma recursiva. El algoritmo multiplica `a` por sí mismo `b-1` veces, reduciendo `b` en cada llamada.

## Número de caminos en una cuadrícula (Recursividad)

Calcula cuántos caminos posibles existen desde la esquina superior izquierda hasta la esquina inferior derecha de una cuadrícula, moviéndose solo hacia la derecha o hacia abajo. El algoritmo suma los caminos desde la celda a la derecha y la celda abajo.

## Invertir una cadena (Recursividad)

Invierte una cadena de texto utilizando un enfoque recursivo. El algoritmo toma el último carácter de la cadena y lo concatena a la inversión del resto de la cadena.

## Torres de Hanói (Recursividad)

Resuelve el problema de las Torres de Hanói moviendo discos de la torre de origen a la torre destino utilizando una torre auxiliar. El algoritmo mueve discos recursivamente entre las torres siguiendo las reglas del juego.

## Contar ocurrencias de un número en una lista (Recursividad)

Cuenta cuántas veces aparece un número en una lista de forma recursiva. El algoritmo verifica el primer elemento y suma 1 si coincide, luego llama a sí mismo con el resto de la lista.

## Sumar dígitos de un número (Recursividad)

Suma los dígitos de un número utilizando un método recursivo. El algoritmo toma el último dígito y lo suma a la suma de los dígitos del número sin ese último dígito.

## Ordenamiento por selección

Implementa el algoritmo de ordenamiento por selección para ordenar una lista. El algoritmo encuentra el elemento más pequeño en la lista y lo intercambia con el primer elemento, repitiendo el proceso para la sublista restante.

## Ordenamiento Burbuja

Completa el algoritmo de ordenamiento burbuja para ordenar una lista de elementos. El algoritmo compara elementos adyacentes y los intercambia si están en el orden incorrecto, repitiendo el proceso hasta que la lista esté ordenada.

## Ordenamiento por Inserción

Implementa el algoritmo de ordenamiento por inserción para organizar una lista. El algoritmo toma un elemento de la lista y lo coloca en la posición correcta en la parte ya ordenada, repitiendo el proceso para todos los elementos.

## Ordenamiento Rápido (Quicksort)

Completa la implementación del algoritmo de ordenamiento rápido para ordenar una lista eficientemente. El algoritmo selecciona un pivote y particiona la lista en elementos menores y mayores que el pivote, ordenando recursivamente las sublistas.

## Ordenamiento por Mezcla (Merge Sort)

Implementa el algoritmo de ordenamiento por mezcla para dividir y conquistar al ordenar una lista. El algoritmo divide la lista en dos mitades, las ordena recursivamente y luego las combina en una lista ordenada.

## Comparación entre Burbuja y Selección

Mide el tiempo de ejecución de los algoritmos de ordenamiento burbuja y selección para comparar su rendimiento. El algoritmo registra el tiempo que toma cada método para ordenar la misma lista y compara los resultados.

## Ordenar una Lista de Cadenas por Longitud

Ordena una lista de cadenas basándose en la longitud de cada cadena. El algoritmo utiliza un método de ordenamiento, como el burbuja o selección, adaptado para comparar longitudes.

## Búsqueda Secuencial en una Lista

Implementa una búsqueda secuencial para encontrar un elemento en una lista no ordenada. El algoritmo recorre la lista elemento por elemento hasta encontrar el buscado o llegar al final.

## Búsqueda Binaria en una Lista Ordenada

Completa el código para realizar una búsqueda binaria en una lista que ya está ordenada. El algoritmo divide la lista y compara el elemento medio con el buscado, ajustando el rango de búsqueda según el resultado.

## Búsqueda de Máximo y Mínimo en una Lista

Encuentra el número máximo y mínimo en una lista utilizando un enfoque recursivo. El algoritmo compara el primer elemento con el máximo y mínimo de la sublista restante, actualizándolos según sea necesario.