

Análisis y Diseño de Algoritmos

Ejercicios Tema 4

Francisco Javier Mercader Martínez

- 1) Diseñar un algoritmo para calcular el mayor y el segundo mayor elemento de un array de enteros utilizando la técnica divide y vencerás.

Calcular el número de comparaciones realizadas en el peor y el mejor caso suponiendo n potencia de 2.

¿Sería el orden obtenido extrapolable a un n que no sea potencia de 2?

- 1) Dividir el problema: Dividimos el array en dos mitades iguales de tamaño $\frac{n}{2}$.
- 2) Resolver el subproblema: Recursivamente encontramos el mayor y el segundo mayor en cada mitad.
- 3) Combinar los resultados:
 - Comparar los dos elementos máximos obtenidos de las mitades para determinar el máximo global (**max1**).
 - El segundo mayor (**max2**) será el mayor entre el segundo mayor de la mitad que contiene al máximo global y el máximo de la otra mitad.

```
def encontrar_mayores(arr):  
    """  
    Encuentra el mayor y el segundo mayor elemento de un arreglo utilizando divide y  
    vencerás.  
  
    - arr: Lista de enteros.  
    Retorna: (mayor, segundo_mayor, comparaciones)  
    """  
    def dividir_y_vencer(arr):  
        # Caso base: Si hay solo dos elementos, compara directamente  
        if len(arr) == 2:  
            if arr[0] > arr[1]:  
                return arr[0], arr[1], 1 # mayor, segundo mayor, comparaciones  
            else:  
                return arr[1], arr[0], 1  
  
        # Divide el arreglo en dos mitades  
        mid = len(arr) // 2  
        izq_mayor, izq_segundo, izq_comparaciones = dividir_y_vencer(arr[:mid])  
        der_mayor, der_segundo, der_comparaciones = dividir_y_vencer(arr[mid:])  
  
        # Combina las soluciones  
        comparaciones = izq_comparaciones + der_comparaciones  
  
        if izq_mayor > der_mayor:  
            mayor = izq_mayor  
            segundo_mayor = max(izq_segundo, der_mayor)  
        else:  
            mayor = der_mayor
```

```

        segundo_mayor = max(der_segundo, izq_mayor)

        comparaciones += 2 # Comparaciones para determinar mayor y segundo mayor
        return mayor, segundo_mayor, comparaciones

# Llamar a la función recursiva
return dividir_y_vencer(arr)

# Ejemplo de uso
if __name__ == '__main__':
    arr = [10, 3, 5, 7, 9, 2, 6, 8]
    mayor, segundo_mayor, comparaciones = encontrar_mayores(arr)
    print(f"Mayor elemento: {mayor}")
    print(f"Segundo mayor elemento: {segundo_mayor}")
    print(f"Total de comparaciones: {comparaciones}")

```