

# Examen 2

## Análisis y Diseño de Algoritmos

21 de Octubre del 2021

**Ejercicio 1 (4.5 ptos).** Considere el siguiente problema.

Entrada: Un vector  $A[1..n]$  **de ceros y unos** tal que todos los ceros ocurren antes que todos los unos. Salida: La posición del primer 1. Por ejemplo, si  $A = [0, 0, 0, 0, 1, 1]$ , la respuesta es 5. Considere que el arreglo tiene por lo menos un 1.

Diseñe un algoritmo de **división y conquista** (escriba el pseudocódigo del algoritmo)  $\Theta(\lg n)$  en el peor caso. Escriba una recurrencia para el desempeño de este algoritmo. Compruebe el crecimiento  $\Theta$  de esta recurrencia **usando teorema maestro**. *Además, use un ejemplo sencillo explicando brevemente cómo su algoritmo funciona para dicho caso en particular.*

**Ejercicio 2 (3 ptos).** Pruebe por inducción que la altura de un nodo  $i$  de un heap  $A[1..n]$  es menor o igual que  $\lg \frac{n}{i}$ . Puede saber que es conocido cuanto es el valor de la altura de un heap con  $n$  nodos (no necesita probar este hecho).

**Ejercicio 3 (3.5 ptos).** Considere el arreglo  $A_1 = [9, 13, 5, 12, 8, 7]$ . Ilustre el algoritmo BUILD-MAX-HEAP en dicho arreglo. Sea  $A_2$  el arreglo resultante. Ilustre el algoritmo HEAP-EXTRACT-MAX en el arreglo  $A_2$ . Luego de esta operación, ilustre el algoritmo HEAP-SORT en el arreglo  $A_2$  (que acaba de ser modificado). Debe mostrar todos los pasos en su ilustración.

**Ejercicio 4 (4 ptos).** Sean  $A[1..n], B[1..n]$  dos arreglos de 0s y 1s (es decir, cada uno de sus elementos o es 0 o es 1). El siguiente algoritmo determina si existe una posición  $i$ , tal que  $A[i] = B[i] = 1$ .

ALGO( $A, B, n$ )

```
1: for  $i = 1$  to  $n$ 
2:   if ( $A[i] == B[i] == 1$ )
3:     return true
4: return false
```

Considere las siguientes variables aleatorias.

$$X_i = \begin{cases} i & : \text{la primera coincidencia pedida ocurre en la posición } i \\ 0 & : \text{caso contrario} \end{cases}$$

(a) Para cada  $i$ , halle  $E[X_i]$ .

- (b) Use el item (a) para encontrar el tiempo esperado de ejecución del algoritmo. Puede suponer que es conocido que  $\sum_{k=0}^{\infty} kx^k = \frac{x}{(1-x)^2}$  cuando  $|x| < 1$ .

**Ejercicio 5 (3.5 ptos).** ¿Cuántos intercambios (entre elementos) hace el Quicksort (usando el pseudocódigo visto en clase, cuando recibe un arreglo con  $n$  elementos iguales? Debe dar el número exacto. Justifique.

**Ejercicio 6 (2 ptos).** Esta parte es electiva. **Responda solo una de las siguientes preguntas (si responde más de una no se le considerará puntaje).**

- En el algoritmo de división y conquista para calcular las inversiones. Demuestre que, luego de terminar las llamadas recursivas, si  $(i, j)$  es una inversión con  $i \leq \lfloor (p+r)/2 \rfloor < j$ , entonces  $(i', j')$  es también una inversión si  $i < i' \leq \lfloor (p+r)/2 \rfloor$  y  $\lfloor (p+r)/2 \rfloor \leq j' < j$ .
- Indique cuales son las tres llamadas recursivas en el algoritmo de Karatsuba y explique por qué es suficiente hacer estas únicas tres llamadas.