

# Examen 2

## Análisis y Diseño de Algoritmos

26 de Mayo del 2020

**Ejercicio 1 (2.5 ptos).** ¿Cuántas hojas tiene un heap con  $n$  nodos? Demuestre.

**Ejercicio 2 (4 ptos).** Considere el siguiente algoritmo que determina el segundo mayor elemento de un vector  $v[1 \dots n]$  con  $n \geq 2$  números positivos distintos.

ALGO( $v, n$ )

```
1:  $mayor = 0$ 
2:  $segundo\_mayor = 0$ 
3: for  $i = 1$  to  $n$ 
4:   if  $v[i] > mayor$ 
5:      $segundo\_mayor = mayor$ 
6:      $mayor = v[i]$ 
7:   else
8:     if  $v[i] > segundo\_mayor$ 
9:        $segundo\_mayor = v[i]$ 
10: return  $segundo\_mayor$ 
```

Suponga que  $v$  es una permutación de 1 a  $n$  escogida de entre todas las permutaciones de 1 a  $n$  con distribución uniforme de probabilidad. Sea  $X$  la variable aleatoria que guarda el número de veces que la variable  $segundo\_mayor$  es alterada (osea, el número de ejecuciones de las líneas 5 y 9 del algoritmo) en una llamada a ALGO( $v, n$ ). Calcule el valor esperado de  $X$ .

**Ejercicio 3 (6 ptos).** Dada una matriz  $A[1..n, 1..m]$  ( $n$  filas y  $m$  columnas) de números reales, un *retazo* de  $A$  es un arreglo  $r[1..n]$  tal que  $r[i] \in \{1, \dots, m\}$  y  $|r[i] - r[i-1]| \leq 1$  para todo  $i > 1$ .

Por ejemplo, si

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 4 \\ 4 & 1 & 2 \end{bmatrix},$$

ejemplos de retazos en  $A$  son  $[1, 2, 3]$ ,  $[2, 1, 3]$ . Pero el siguiente arreglo no es un retazo en  $A$ :  $[1, 3, 2]$ .

El *peso* de un retazo  $r$  es la suma de los correspondientes valores en  $A$ . En el ejemplo anterior, el peso del retazo  $[2, 1, 3]$  es  $A[1, 2] + A[2, 1] + A[3, 3] = 2 + 3 + 2 = 7$ .

Considere el siguiente problema.

**Problema Min-Retazo.** Dada una matriz, encontrar un retazo de peso mínimo.

- (a) Sea  $OPT(i, j)$  el valor de un retazo de peso mínimo en la matriz  $A[1..i, 1..m]$  que termina en  $(i, j)$ . Escriba una recurrencia para  $OPT(i, j)$ .
- (b) Escriba un algoritmo de programación dinámica para el problema MIN-RETAZO a partir de su recurrencia.

**Ejercicio 4 (3 ptos).** ¿Cual es el tiempo de ejecución de Quicksort en el peor caso? ¿Cual es el tiempo de ejecución de Quicksort en el mejor caso? Para ambas situaciones, de un ejemplo de un arreglo de 7 elementos.

**Ejercicio 5 (4.5 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  $A_2$ . Ilustre el algoritmo HEAP-SORT en el arreglo  $A_2$ .