



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACION

Diseño y Análisis de Algoritmos - IIC2283
Examen

El plazo de entrega del examen es el domingo 13 de diciembre a la 1pm. Para entregar el examen debe utilizar el mismo repositorio de git que usted usó para entregar las interrogaciones 1 y 2. El examen tiene 4 preguntas; las soluciones de estas preguntas deben ser colocadas en el directorio **examen** de su repositorio con los nombres **p1.ext**, **p2.ext**, **p3.ext** y **p4.ext**, donde **ext** es la extensión correspondiente al tipo de archivos que usted suba en su repositorio (se recomienda utilizar **pdf** o **jpeg**). Las respuestas pueden ser escritas a mano o en la Latex (no se recomienda el uso de Word, pero no está prohibido). Para realizar consultas sobre la interrogación utilice el foro <https://github.com/PUC-IIC2283/2020-IIC2283-Pruebas/issues>.

Importante: en esta prueba puede utilizar los apuntes del curso.

1. [1.5 puntos] Considere la siguiente ecuación de recurrencia:

$$T(n) = \begin{cases} 1 & n = 0 \\ T(\lfloor \frac{5n}{6} \rfloor) + T(\lfloor \frac{6n}{7} \rfloor) + T(\lfloor \frac{7n}{8} \rfloor) + n^9 & n > 0 \end{cases}$$

Encuentre una función $f(n)$ tal que $T(n) \in \Theta(f(n))$, y justifique por qué $f(n)$ satisface esta condición.

2. Suponga que tiene un procedimiento **LanzarMonedaCargada()** que retorna 0 con probabilidad p y 1 con probabilidad $1 - p$. Utilizando este procedimiento, el objetivo de esta pregunta es construir un algoritmo **LanzarMoneda()** que retorna 0 con probabilidad $\frac{1}{2}$ y 1 con probabilidad $\frac{1}{2}$, vale decir, un algoritmo que implementa una moneda no cargada.

Considere el siguiente algoritmo de tipo Las Vegas para resolver el problema:

```
LanzarMoneda()  
   $x := \text{LanzarMonedaCargada}()$   
   $y := \text{LanzarMonedaCargada}()$   
  if  $x \neq y$  then return  $x$   
  else return sin_resultado
```

- (a) [0.5 puntos] Dado $b \in \{0, 1\}$, demuestre que:

$$\Pr(\text{LanzarMoneda()} \text{ retorna } b \mid \text{LanzarMonedaCargada()} \text{ retorna } 0 \text{ ó } 1) = \frac{1}{2}.$$

- (b) [0.5 puntos] Encuentre el valor $\Pr(\mathbf{LanzarMoneda}())$ retorna 0 ó 1.
- (c) [0.5 puntos] Indique cuántas veces se debe ejecutar **LanzarMoneda()** en promedio para obtener el valor 0 ó 1.
3. [1.5 puntos] Considere el siguiente algoritmo que recibe como entrada números $n, k \in \mathbb{N}$ tales que $n = 4 \cdot \ell + 3$ y $k \geq 1$:

```

TestPrimalidad( $n, k$ )
  if EsPotencia( $n$ ) then return COMPUESTO
  else
    sea  $a_1, \dots, a_k$  una secuencia de números elegidos de
                                manera uniforme e independiente desde  $\{1, \dots, n-1\}$ 
    for  $i := 1$  to  $k$  do
      if MCD( $a_i, n$ )  $> 1$  then return COMPUESTO
      else  $b_i := \mathbf{EXP}(a_i, \frac{n-1}{2}, n)$ 
    for  $i := 1$  to  $k$  do
      if  $b_i \not\equiv 1 \pmod{n}$  and  $b_i \not\equiv -1 \pmod{n}$  then return COMPUESTO
    return PRIMO

```

Demuestre que **TestPrimalidad**(n, k) verifica si n es un número primo con una probabilidad de error menor o igual a $(\frac{1}{2})^k$.

4. [1.5 puntos] Dado $n \in \mathbb{N}$, defina $\mathcal{E}_n$ como el conjunto de listas L con n elementos distintos sacados desde el conjunto $\{1, \dots, n\}$. Dada una lista $L \in \mathcal{E}_n$, defina la siguiente medida de cuán desordenada está la lista L :

$$\mathbf{PosCam}(L) = |\{(i, j) \mid i, j \in \{1, \dots, n\}, i < j \text{ y } L[i] > L[j]\}|.$$

Por ejemplo, si $L = [1, 2, \dots, n-1, n]$ es una lista ordenada, se tiene que $\mathbf{PosCam}(L) = 0$, y si $L = [n, n-1, \dots, 2, 1]$ es una lista invertida, se tiene que $\mathbf{PosCam}(L) = \frac{n \cdot (n-1)}{2}$.

Suponga que tiene la distribución uniforme sobre la listas en $\mathcal{E}_n$, y defina la variable aleatoria X_n como $X_n(L) = \mathbf{PosCam}(L)$. En esta pregunta usted debe calcular $E(X_n)$.

Pista: esta pregunta puede ser resuelta con las técnicas utilizadas para analizar el algoritmo **Quicksort** en el caso promedio.