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

Diseño y Análisis de Algoritmos - IIC2283
Examen
Tiempo: 2.5 horas

1. En esta pregunta usted debe demostrar y generalizar el Teorema Chino del Resto.

   (a) [1 punto] Suponga que m, n son dos números naturales tales que $m \geq 2$, $n \geq 2$ y $\text{MCD}(n, m) = 1$. Demuestre que para cualquier par de números naturales a y b, existe un número c tal que:

   $$\begin{aligned} c &\equiv a \mod m \\ c &\equiv b \mod n \end{aligned}$$

   (b) [1 punto] Suponga que $n_1, \ldots, n_k$ es una secuencia de números naturales tales que $k \geq 2$, $n_i \geq 2$ para cada $i \in \{1, \ldots, k\}$, y $\text{MCD}(n_i, n_j) = 1$ para cada $i, j \in \{1, \ldots, k\}$ tales que $i \neq j$. Demuestre que para cualquier secuencia $a_1, \ldots, a_k$ de números naturales, existe un número c tal que para cada $i \in \{1, \ldots, k\}$:

   $$c \equiv a_i \mod n_i$$

2. Sean p, q dos números primos distintos tales que $p \geq 3$ y $q \geq 3$, y sea $n = p \cdot q$.

   (a) [1 punto] Demuestre que para cada $a \in \mathbb{Z}_n^*$, se cumple que a tiene raíz cuadrada en módulo n si y sólo si a tiene raíz cuadrada tanto en módulo p como en módulo q, es decir,

   $$\begin{gathered} \text{existe } b \text{ tal que } b^2 \equiv a \bmod n \\ \text{si y sólo si} \\ \text{existen } c \text{ y } d \text{ tales que } c^2 \equiv a \bmod p \text{ y } d^2 \equiv a \bmod q. \end{gathered}$$

   (b) [1.5 puntos] De un algoritmo aleatorizado de Las Vegas que calcule raíces cuadradas en módulo n. En particular, utilizando la notación desarrollada en clases de un pseudocódigo para una función **RaízCuadrada**(p, q, a, k), la cual recibe cómo parámetros dos números primos distintos p y q tales que $p \geq 3$ y $q \geq 3$, un número natural $a \in \mathbb{Z}_n^*$ con $n = p \cdot q$ y un número natural $k \geq 1$, y retorna uno de los siguientes valores: si a no tiene raíz cuadrada de módulo n entrega *sin_raíz_cuadrada*, y si a tiene raíz cuadrada en módulo n, entonces con probabilidad $(1 - (\frac{3}{4})^k)$ entrega un número b tal que $b^2 \equiv a \bmod n$, y con probabilidad $(\frac{3}{4})^k$ entrega *sin_resultado*.

Analice la complejidad del algoritmo considerando como operaciones básicas a contar la suma, resta, multiplicación, división y cálculo del resto para números enteros, y muestre que funciona en tiempo polinomial en el peor caso. Además, demuestre que la probabilidad de que el algoritmo entregue *sin_resultado* es $(\frac{3}{4})^k$.

Importante: Para resolver esta pregunta puede utilizar todos los resultados en las transparencias de clases. Si quiere utilizar el algoritmo visto en la tarea 2 lo tiene que describir de manera completa, no puede simplemente decir que existe un algoritmo para calcular raíces cuadradas en módulo un número primo.

3. [1.5 puntos] Una fórmula en 3-CNF es una expresión de la lógica proposicional de la forma:

$$(\ell_{1,1} \vee \ell_{1,2} \vee \ell_{1,3}) \wedge (\ell_{2,1} \vee \ell_{2,2} \vee \ell_{2,3}) \wedge \cdots \wedge (\ell_{n,1} \vee \ell_{n,2} \vee \ell_{n,3}),$$

donde cada $\ell_{i,j}$ ($1 \leq i \leq n$ y $1 \leq j \leq 3$) es un literal, vale decir, una variable proposicional o la negación de una variable proposicional. Por ejemplo, $(p \vee \neg q \vee r) \wedge (\neg p \vee r \vee \neg s) \wedge (q \vee r \vee s)$ es una fórmula en 3-CNF. Cada expresión de la forma $(\ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3})$ se denomina una cláusula. Para propósitos de este problema suponemos que cada fórmula en 3-CNF cumple lo siguiente: (a) ninguna cláusula contiene una variable proposicional y su negación, y (b) cada cláusula contiene exactamente tres literales distintos. Además, dada una fórmula φ en 3-CNF, definimos $\text{var}(\varphi)$ como el conjunto de variables mencionadas en φ. Por ejemplo, $\text{var}((p \vee \neg q \vee r) \wedge (\neg p \vee r \vee \neg s) \wedge (p \vee r \vee s)) = \{p, q, r, s\}$.

Definimos MAX-3-CNF-SAT como el problema de encontrar una valuación para las variables proposicionales de una fórmula en 3-CNF que maximice el número de cláusulas que se satisfacen. Considere un algoritmo aleatorizado para este problema que, dada una fórmula φ en 3-CNF tal que $\text{var}(\varphi) = \{x_1, \ldots, x_k\}$, asigna de manera independiente y con distribución uniforme valores a x_1, ..., x_k, es decir, simplemente asigna a cada variable proposicional x_i un 1 con probabilidad $\frac{1}{2}$ y un 0 con probabilidad $\frac{1}{2}$. Demuestre que el número esperado de cláusulas satisfechas después de ejecutar este algoritmo en una fórmula con n cláusulas es $\frac{7n}{8}$.