MATEMÁTICA DISCRETA (Grado en Ingeniería Informática)
CURSO 2018/19. Convocatoria Ordinaria 1.

Nombre:______________________________ DNI:________Gr. Teoría:__ Gr. Práct.:__

Evaluación Continua	Lógica. Nota:_________	Prácticas:	Apto. Nota________
	Asistencia a complejidad		Ordinaria 1__________

1.- [10 *puntos*] Definir forma argumentativa inválida y usar el método de refutación (justificando cada paso necesario) para estudiar la validez o invalidez de la siguiente argumentación:

$$\mathcal{A}_1: (\sim p_1) \leftrightarrow (p_2 \vee p_5),$$
$$\mathcal{A}_2: \sim((p_7 \downarrow p_9) \rightarrow (p_3 \rightarrow (\sim p_8)));$$
$$\therefore \mathcal{A}: ((p_4 \leftrightarrow p_9) \wedge (p_9 \oplus p_4)) \uparrow p_6$$

2.- [*10 puntos*] Sea $X = \{x \in \mathbb{Z}$: x es par$\}$ y en él se define la siguiente relación binaria:

$$x\ R\ y \text{ si y sólo si } x + y \equiv 0 \bmod 4$$

a) [*4 puntos*] Estudiar si R es una relación de equivalencia en X.
b) [6 *puntos*] Definir conjunto cociente y clase de equivalencia. Calcular, si es posible, X/R.

3.- [*10 puntos*]: Dada la función booleana $f: (\mathbb{B}_2)^3 \longrightarrow \mathbb{B}_2$ definida por $f(x, y, z) = (\overline{z} \rightarrow y) \oplus (\overline{y} \uparrow z)$.
Se pide:
i) [6 *puntos*] Calcular la expresión en maxtérminos de la función.
ii) [4 *puntos*] Calcular el polinomio de Gegalkine de f.

4.- [*10 puntos*]
a) [*1 punto*] Enunciar el teorema que caracteriza la resolución de una ecuación diofántica.
b) [*5 puntos*] Utilizar dicho teorema para calcular, si es posible, todas las soluciones enteras de

$$23\,x \equiv -2 \bmod 51$$

c) [*2 puntos*] Utilizar el teorema del apartado a) para calcular, si es posible, todas las soluciones enteras de

$$6\,x \equiv 5 \bmod 100$$

d) [*2 puntos*] Utilizar el teorema del apartado a) para caracterizar todas las soluciones enteras de

$$a\,x \equiv c \bmod n$$

5.- [10 *puntos*]
a) [*3 puntos*] Explicar qué determina el siguiente algoritmo y aplicarlo para la función $f: A \rightarrow \mathbb{Z}_3$ dada por $f(x) = \overline{x+2}$ donde A es el conjunto de los tres primeros primos positivos.

PROGRAMA
A = {LISTA DE ELEMENTOS DEL DOMINIO}; B = {LISTA DE ELEMENTOS DEL CODOMINIO}; DEFINICIÓN DEL GRAFO DE LA APLICACIÓN;
variable= {}; Do[variable = Union[variable, Append[{}, f[A[[i]]]]], {i, 1, Length[A]}]; variable

b) [*4 puntos*] Definir complejidad en tiempo y calcularla para el algoritmo.

c) [*3 puntos*] Añadir código, al final del programa, para que el programa anterior devuelva si f es o no sobreyectiva.