



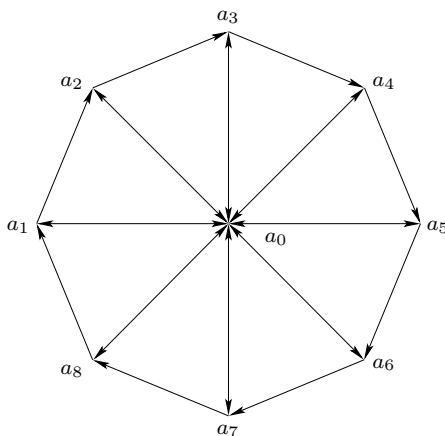
PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
 ESCUELA DE INGENIERIA
 DEPARTAMENTO DE CIENCIA DE LA COMPUTACION

Matemáticas Discretas - IIC1253
Examen

1. [1.2 puntos] Un grafo dirigido $G = (N, A)$ es una k -rueda si se cumple que:

$$\begin{aligned} N &= \{a_0, a_1, \dots, a_k\}, \\ A &= \{(a_i, a_{i+1}) \mid i \in \{1, \dots, k-1\}\} \cup \{(a_k, a_1)\} \cup \{(a_0, a_i), (a_i, a_0) \mid i \in \{1, \dots, k\}\}. \end{aligned}$$

Por ejemplo, el siguiente grafo es una 8-rueda:



Además, un grafo $G' = (N', A')$ es un *subgrafo* de un grafo $G = (N, A)$ si $N' \subseteq N$ y $A' \subseteq A \cap (N' \times N')$. Suponga que $G = (N, A)$, donde $N = \{1, \dots, n\}$ y $n \geq 2$, y suponga que k es un número natural tal que $1 \leq k < n$. Construya una fórmula φ en lógica proposicional tal que φ es satisfacible si y sólo si G tiene un subgrafo G' tal que G' es una k -rueda.

Importante: La fórmula φ construida en esta pregunta debe ser de tamaño polinomial en n , vale decir, debe existir un polinomio $p(x)$ tal que $\text{la}(\varphi) \leq p(n)$.

2. Sea $\mathcal{L} = \{<\}$, y sea Σ un conjunto formado por las siguientes $\mathcal{L}$ -oraciones:

$$\begin{aligned} \varphi_1 &= \forall x \neg(x < x) \\ \varphi_2 &= \forall x \forall y \forall z ((x < y \wedge y < z) \rightarrow x < z) \\ \varphi_3 &= \forall x \forall y (x < y \vee y < x \vee x = y). \end{aligned}$$

Conteste las siguientes preguntas.

- (a) [0.3 puntos] Construya una $\mathcal{L}$ -estructura $\mathfrak{A}_1$ tal que $\mathfrak{A}_1 \models \Sigma$.
(b) [0.3 puntos] Construya una $\mathcal{L}$ -estructura $\mathfrak{A}_2$ tal que $\mathfrak{A}_2 \models \Sigma \cup \{\varphi_4, \varphi_5, \varphi_6\}$, donde:

$$\begin{aligned}\varphi_4 &= \forall x \exists y (x < y) \\ \varphi_5 &= \forall x \exists y (y < x) \\ \varphi_6 &= \forall x \forall y (x < y \rightarrow \exists z (x < z \wedge z < y)).\end{aligned}$$

- (c) [0.3 puntos] Construya una $\mathcal{L}$ -estructura $\mathfrak{A}_3$ tal que $\mathfrak{A}_3 \models \Sigma \cup \{\varphi_6, \varphi_7, \varphi_8\}$, donde φ_6 es la $\mathcal{L}$ -oración definida en (b) y:

$$\begin{aligned}\varphi_7 &= \exists x \forall y (y < x \vee y = x) \\ \varphi_8 &= \exists x \forall y (x < y \vee y = x)\end{aligned}$$

- (d) [0.3 puntos] Construya una $\mathcal{L}$ -estructura $\mathfrak{A}_4$ tal que $\mathfrak{A}_4 \models \Sigma \cup \{\varphi_7, \varphi_8, \varphi_9\}$, donde φ_7 y φ_8 son las $\mathcal{L}$ -oraciones definidas en (c) y:

$$\varphi_9 = \forall x (\exists y (x < y) \rightarrow \exists z (x < z \wedge \neg \exists w (x < w \wedge w < z))).$$

3. [1.2 puntos] Sea $\{0, 1\}^\omega$ el conjunto de los strings infinitos de la forma $a_0 a_1 a_2 a_3 \dots$, donde cada a_i ($i \in \mathbb{N}$) es 0 ó 1. Demuestre que $\{0, 1\}^\omega$ es equinumeroso con $2^{\mathbb{N}}$.
4. Sea $n \geq 3$ un número natural fijo. Considere el siguiente algoritmo para calcular $2^a \bmod n$ para $a \geq 1$:

```

expn(a)
  if  $a = 1$  then return 2
  else if  $a \bmod 2 = 0$  then
     $t = \mathbf{exp}_n(\frac{a}{2})$ 
     $r = (t \cdot t) \bmod n$ 
    return  $r$ 
  else
     $t = \mathbf{exp}_n(\frac{a-1}{2})$ 
     $r = (t \cdot t \cdot 2) \bmod n$ 
    return  $r$ 

```

Conteste las siguientes preguntas.

- (a) [0.6 puntos] Sea $T(a)$ el número de multiplicaciones, divisiones y cálculos de módulo que hace el algoritmo **exp**_n para calcular $2^a \bmod n$ (en el peor caso). Determine una ecuación de recurrencia para T .
(b) [0.6 puntos] Utilizando un reemplazo de variable encuentre una solución para la ecuación de recurrencia. Use este resultado para determinar una función f tal que $T(a) \in O(f(a))$, y justifique por qué $T(a) \in O(f(a))$.
Importante: El puntaje asignado a su solución dependerá de qué tan buena es la cota superior encontrada.
5. [1.2 puntos] Sea $p \geq 3$ un número primo fijo. Utilizando el algoritmo definido en la pregunta 4, desarrolle un algoritmo **dexp**_p para calcular:

$$2^{(2^a)} \bmod p$$

En particular, usted debe demostrar que si $T'(a)$ es el número de multiplicaciones, divisiones y cálculos de módulo que hace el algoritmo **dexp**_p para calcular $2^{(2^a)} \bmod p$ (en el peor caso), entonces $T'(a) \in O(f(a))$, donde f es la función mencionada en la pregunta 4 (b).