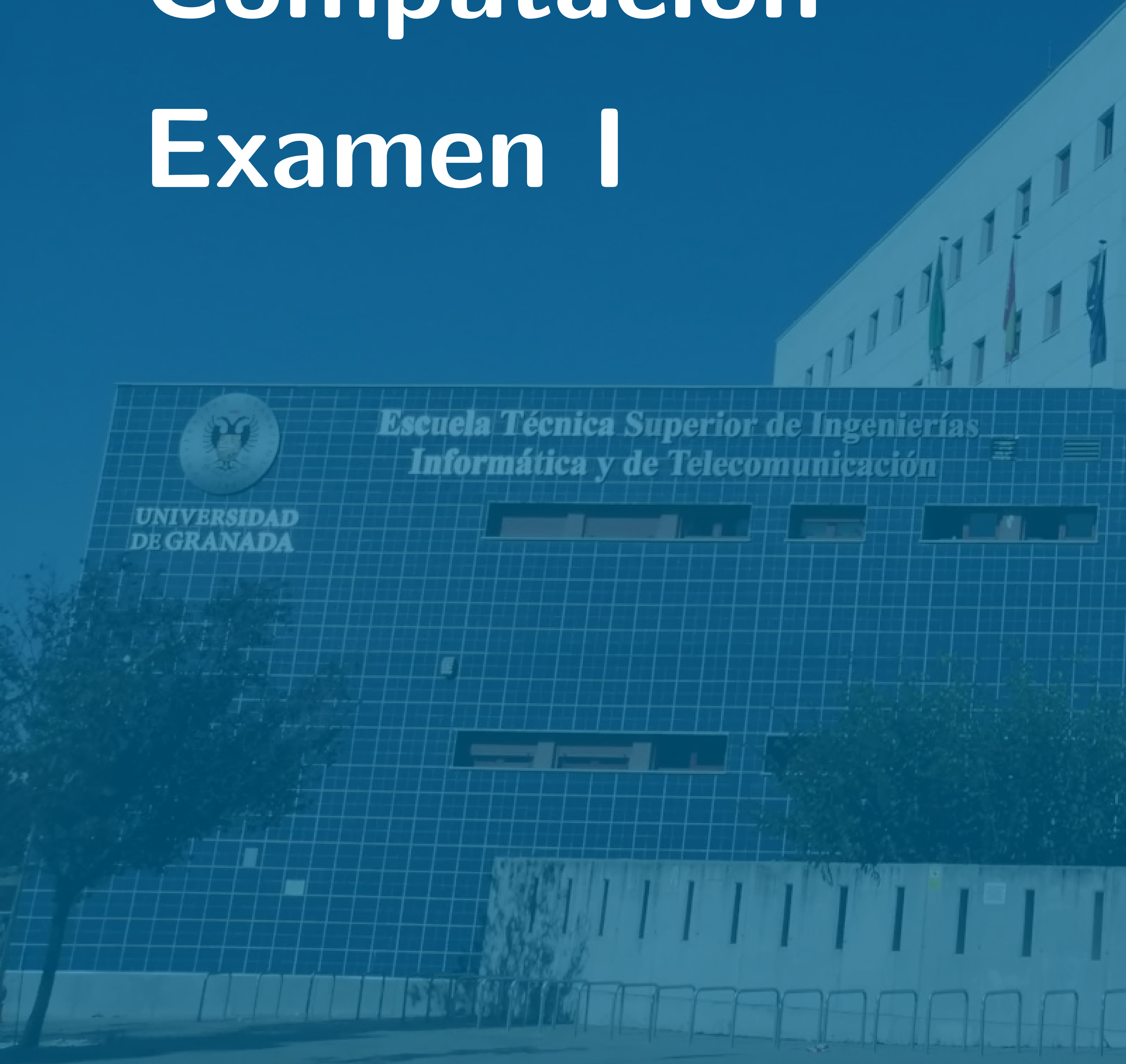
Modelos de Computación
Examen I



Los Del DGIIM, `losdeldgiim.github.io`

Doble Grado en Ingeniería Informática y Matemáticas
Universidad de Granada

Modelos de Computación Examen I

Los Del DGIIM, `losdeldgiim.github.io`

Arturo Olivares Martos

Granada, 2024-2025

Asignatura Modelos de Computación

Curso Académico 2021-22.

Grado Doble Grado en Ingeniería Informática y Matemáticas.

Descripción Parcial Tema 1.

Ejercicio 1. Dar gramáticas que acepten los siguientes lenguajes:

1. $L_1 = \{a^n b^m c^k \in \{a,b,c\}^* \mid n,m,k \in \mathbb{N} \cup \{0\}, |n-m| = k\}$.

   Tenemos que $L_1 = L_{11} \cup L_{12}$, donde:

   - $L_{11} = \{a^n b^m c^k \in \{a,b,c\}^* \mid n,m,k \in \mathbb{N} \cup \{0\}, n = m + k\}$.
   - $L_{12} = \{a^n b^m c^k \in \{a,b,c\}^* \mid n,m,k \in \mathbb{N} \cup \{0\}, m = n + k\}$.

   Sea $G_1 = (\{S_1, X\}, \{a,b,c\}, P_1, S_1)$ la gramática que genera L_{11}, donde P_1 es el conjunto de reglas de producción:

   $$\begin{aligned} S_1 &\to aSc \mid X \\ X &\to aXb \mid \varepsilon \end{aligned}$$

   Sea $G_2 = (\{S_2, Y_1, Y_2\}, \{a,b,c\}, P_2, S_2)$ la gramática que genera L_{12}, donde P_2 es el conjunto de reglas de producción:

   $$\begin{aligned} S_2 &\to Y_1 Y_2 \\ Y_1 &\to aY_1 b \mid \varepsilon \\ Y_2 &\to bY_2 c \mid \varepsilon \end{aligned}$$

   Tenemos que $L_{11} = \mathcal{L}(G_1)$ y $L_{12} = \mathcal{L}(G_2)$. Sea entonces G la gramática dada por $G = (\{S, S_1, S_2, X, Y_1, Y_2\}, \{a,b,c\}, P, S)$ la gramática que genera L_1, donde P es el conjunto de reglas de producción:

   $$\begin{aligned} S &\to S_1 \mid S_2 \\ S_1 &\to aSc \mid X \\ X &\to aXb \mid \varepsilon \\ S_2 &\to Y_1 Y_2 \\ Y_1 &\to aY_1 b \mid \varepsilon \\ Y_2 &\to bY_2 c \mid \varepsilon \end{aligned}$$

   Tenemos de forma directa que $L_1 = \mathcal{L}(G)$.

2. $L_2 = \{u \in \{0,1\}^* \mid n_0(u),\ n_1(u) \text{ es par}\}$.

   Consideramos los siguientes estados:

   - S: Palabra leída correcta, con un número par de 0's y de 1's.
   - E_0: El error está en el número de 0's, ya que este es impar.
   - E_1: El error está en el número de 1's, ya que este es impar.
   - E_{01}: El error está en el número de 0's y de 1's, ya que ambos son impares.

La gramática por tanto que genera L_2 es $G = (\{S, E_0, E_1, E_{01}\}, \{0, 1\}, P, S)$, donde P es el conjunto de reglas de producción:

$$\begin{aligned} S &\to 0E_0 \mid 1E_1 \mid \varepsilon \\ E_0 &\to 0S \mid 1E_{01} \\ E_1 &\to 1S \mid 0E_{01} \\ E_{01} &\to 0E_1 \mid 1E_0 \end{aligned}$$

Tenemos de forma directa que $L_2 = \mathcal{L}(G)$.

Ejercicio 2. Dar gramáticas que acepten los siguientes lenguajes:

1. $L_1 = \{u \in \{0, 1\}^* \mid u^{-1} = \overline{u}\}$, donde $\overline{u}$ representa el complemento de u, es decir, cambiando 0's por 1's y viceversa.

   Podemos intuir que este lenguaje no es regular, puesto que no se puede reconocer con memoria finita. Es directo ver que, cada vez que introduzcamos un 1 por el inicio, hemos de introducir un 0 por el final, y viceversa. Por tanto, podemos definir la gramática $G = (\{S\}, \{0, 1\}, P, S)$, donde P es el conjunto de reglas de producción:

$$S \to 0S1 \mid 1S0 \mid \varepsilon$$

2. $L_2 = \{a^n b^m \mid n, m \in \mathbb{N} \cup \{0\}, 3m \geqslant n \geqslant 2m\}$.

   En este caso, por cada b que introduzcamos, hemos de introducir al menos 2 o 3 a's. Por tanto, podemos definir la gramática $G = (\{S\}, \{a, b\}, P, S)$, donde P es el conjunto de reglas de producción:

$$S \to aaSb \mid aaaSb \mid \varepsilon$$

Ejercicio 3. Se dejan *propuestos* los siguientes lenguajes, que son más complicados:

1. $L_1 = \{a^n b^m \mid n, m \in \mathbb{N} \cup \{0\}, \text{mcd}(n, m) = 1\}$.

2. L_2, que representa el conjunto de palabras con los paréntesis bien formados.