



Universidad Rey Juan Carlos

Modelos de Examen

Informática Teórica y Lenguajes Formales

Grado en Inteligencia Artificial

**Ana Isabel Gómez Pérez
Francisco Javier Soto Sánchez**

CURSO 2024/2025

©2024 Autor: Ana Isabel Gómez Pérez, Francisco Javier Soto Sánchez. Algunos derechos reservados.

Autoría de transparencias y hojas de ejercicios: Ana Isabel Gómez Pérez. Autoría de videos: Francisco Javier Soto Sánchez

Publicado en : <https://burjcdigital.urjc.es/>

Universidad Rey Juan Carlos, Grado en Inteligencia Artificial
Informática Teórica y Lenguajes Formales
Modelo de Examen I

NOMBRE: _____
DNI: _____

NOTA: Todas las respuestas deben estar etiquetadas y justificadas debidamente para ser válidas..

1. (3 ptos) Dada la siguiente expresión regular, $\alpha = (01^*1)|(10^*0)$
 - (a) (1 punto) Encontrar un autómata finito que acepte el lenguaje generado por la expresión α .
 - (b) (1 punto) Determinizar el autómata obtenido en el apartado anterior.
 - (c) (1 punto) Encontrar una gramática regular que genere el mismo lenguaje.
2. (3 ptos) Dada la siguiente gramática $G = (\{S, A, B, C\}, \{0, 1\}, S, P)$, donde las producciones son,

$$\begin{aligned} S &\mapsto AB \mid 0S1 \mid A \mid CC \\ A &\mapsto 0AB \mid \varepsilon \\ B &\mapsto B1 \\ C &\mapsto B1A \mid C1C0 \end{aligned}$$

Se pide,

- (a) (1,5 ptos) Decir que variables son superfluas. Dar una gramática equivalente y describir el lenguaje generado.
 - (b) (1,5 ptos) Encontrar dos autómatas con pila que acepten el lenguaje generado por la gramática. Mostrar para ambos la computación de la entrada 01, con las transiciones en el espacio de configuraciones
3. (2 ptos) Dado el siguiente lenguaje:

$$L_1 = \{0^n 1^m 2^n, \quad n, m \geq 1\},$$

Se pide

- (a) (1 punto) Dar una gramática libre de contexto sin variables superfluas que genere el lenguaje.
 - (b) (1 punto) Hallar una derivación más a la derecha de la palabra 00011222 utilizando las reglas de shift-reduce y el estado de la pila que seguiría una implementación en SLY.
4. (2 ptos) Se da el siguiente código en lenguaje COOL

```
class main {
main(): object {
  if True==True then if false ==false then 4 else 3
}
}
```

- (a) (1 punto) Corrija libremente el código anterior, manteniendo la funcionalidad, para que sea sintácticamente correcto en el lenguaje COOL .
- (b) (1 punto) Proporcione una regla que permita que el uso de if no requiera else. Explique con un ejemplo si esto crea ambigüedad en la gramática en el contexto de las reglas shift-reduce.

Universidad Rey Juan Carlos, Grado en Inteligencia Artificial
Informática Teórica y Lenguajes Formales
Modelo de Examen I

NOMBRE: _____
DNI: _____

NOTA: Todas las respuestas deben estar etiquetadas y justificadas debidamente para ser válidas..

1. (3 ptos) Dada la siguiente expresión regular, $\alpha = (0|1)^*1(0|1)$
 - (a) (1 punto) Encontrar un autómata finito que acepte el lenguaje generado por la expresión α .
 - (b) (1 punto) Determinizar el autómata obtenido en el apartado anterior.
 - (c) (1 punto) Encontrar una gramática regular que genere el mismo lenguaje.

2. (2 ptos) Dado el siguiente lenguaje:

$$L_1 = \{0^n 1^{m+n} 2^m, \quad n, m \geq 1\},$$

Se pide,

- (a) (1 punto) Dar una gramática libre de contexto sin variables superfluas que genere el lenguaje.
 - (b) (1 punto) Hallar una derivación más a la derecha de la palabra 001112. Las reglas de shift-reduce y el estado de la pila deben seguir las de una implementación en SLY.
3. (3 ptos) Dada la siguiente gramática $G = (\{S, A, B, C\}, \{0, 1, 2\}, S, P)$, donde las producciones son,

$$\begin{aligned} S &\mapsto DBC \mid S0S1 \mid S1S0 \mid A \mid CC \\ A &\mapsto 2ABC \mid \varepsilon \\ B &\mapsto BB \mid \varepsilon \\ C &\mapsto B1AC \mid CDD0 \\ D &\mapsto 1 \mid C1C0 \end{aligned}$$

Se pide,

- (a) (1,5 ptos) Razonar que variables son superfluas. Dar una gramática equivalente y describir el lenguaje generado.
 - (b) (1,5 ptos) Dar gráficamente dos autómatas con pila que acepten el lenguaje generado. Mostrar para ambos la computación de la entrada 01, con las transiciones en el espacio de configuraciones
4. (2 ptos) Se da el siguiente código en lenguaje COOL

```
class main inherits IO{
  main(): Object {{
    a = 0
    while a < 5
      a = a + 1
    pool
    out_string( ' ' Fin de bucle ' '); }} }
```

- (a) (1 punto) Corrija libremente el código anterior, evitando los bucles infinitos, para que sea sintácticamente correcto en el lenguaje COOL.
- (b) (1 punto) Modifique el código para que se imprima en cada iteración el valor de a.

Universidad Rey Juan Carlos, Grado en Inteligencia Artificial
Informática Teórica y Lenguajes Formales
Modelo de Examen II

NOMBRE: _____
DNI: _____

NOTA: Todas las respuestas deben estar etiquetadas y justificadas debidamente para ser válidas..

- (3 pts) Responda a las siguientes cuestiones:
 - (1 punto) Encontrar el autómata finito determinista con el menor número de estados que acepte palabras que no contengan la subpalabra 00
 - (1 punto) Dar una expresión regular que reconozca el lenguaje del apartado anterior.
 - (1 punto) Encontrar el autómata con el menor número de estados que acepta palabras que, o bien el número de 0 en la palabra es impar o no aparece la subpalabra 00. Aclaración: No se pueden cumplir a la vez.
- (2 pts) Dar una gramática equivalente a la siguiente expresión regular, $\alpha = (00|10)(101)^*[01]$. Si una palabra está en $L(\alpha)$ y en $L((101)^*)$, ¿Tiene que ser palíndroma? Razonar la respuesta.
- (3 pts) Dado el siguiente lenguaje:

$$L_1 = \{0^m 1^n 2^{n-m}, n \geq m \geq 1\}$$

Se pide,

- (1,5 pts) Dar una gramática libre de contexto que genere el lenguaje
 - (1,5 pts) Dar dos autómatas con pila que acepten L_1 . El primero tiene que aceptar por pila y cinta vacía y el segundo por estado final.
- (1 punto) Se da el siguiente código en lenguaje COOL

```
class main{ Main(){
(new IO).out string"(Hola "mundo.concat"(\n)) }
}
```

 - (1 punto) Corrija libremente el código anterior para que sea sintácticamente correcto en el lenguaje COOL.
 - (1 punto) Se tiene el siguiente fragmento en el Léxer de SLY

```
TOKEN1=r ' a*(b|c) '
TOKEN2=r ' (ab|c*)d '
TOKEN3=r 'd '
```

 - (0,5 pts) ¿Cuál es la salida del input *adaaabcd* ?
 - (0,5 pts) Proponga un input que pueda ser detectado devolviendo distintos tokens. ¿Cuál se devolvería?

Universidad Rey Juan Carlos, Grado en Inteligencia Artificial
Informática Teórica y Lenguajes Formales
Modelo de Examen II

NOMBRE: _____
DNI: _____

NOTA: Todas las respuestas deben estar etiquetadas y justificadas debidamente para ser válidas..

1. (3 ptos) Se tiene el siguiente fragmento en el Léxer de SLY

TOKEN1= r ' (d | e *) (a | b) * c '
TOKEN2= r ' e * (b | c) * a '
TOKEN3= r ' e | e c '

- (a) (1 punto) Dibuje el autómata finito que se genera como analizador léxico.
(b) (1 punto) Convierta el anterior autómata en uno sin transiciones ϵ .
(c) (1 punto) ¿Qué tokens se devuelven con input *eeceeebbbadc*? Proponga un input que pueda ser detectado devolviendo distintos tokens.
2. (2,5 ptos) Dado el siguiente lenguaje:

$$L_1 = \{0^n 1^{m+n} 2^m, \quad n, m \geq 1\},$$

Se pide,

- (a) (1 punto) Dar una gramática libre de contexto sin variables superfluas que genere el lenguaje.
(b) (1,5 ptos) Dar gráficamente dos autómatas con pila que acepten el lenguaje generado. Mostrar para ambos la computación de la entrada 01, con las transiciones en el espacio de configuraciones
3. (3 ptos) Se da la siguiente gramática libre de contexto $G = (V, \Sigma, S, P)$ donde $V = \{S, X, Y, Z, T\}$, $\Sigma := \{0, 1, 2, 3, +, \times\}$, y las producciones P son:

$$\begin{aligned} S &\mapsto Y \mid Y + S \\ X &\mapsto T \mid Y \times Z \\ Y &\mapsto T \mid (X \times S) \\ Z &\mapsto Z \times Y \mid Y \times Z \\ T &\mapsto 0 \mid 1 \mid 2 \mid 3. \end{aligned}$$

Se pide,

- (a) (1,5 ptos) Razonar que variables son superfluas. Dar una gramática equivalente y describir el lenguaje generado.
(b) (1 punto) Hallar una derivación más a la derecha de la palabra $1 + 2 * 4 + 3$. Las reglas de shift-reduce y el estado de la pila deben seguir las de una implementación en SLY.
4. (2 ptos) Se da el siguiente código en lenguaje COOL

```
class main{ Main(){
(new IO).out string"(Hola "mundo.concat"(\n)) }
}
```

- (a) (1 punto) Corrija libremente el código anterior, evitando los bucles infinitos, para que sea sintácticamente correcto en el lenguaje COOL.
(b) (1 punto) Halle la derivación según la gramática provista siguiendo las reglas de shift reduce en la implementación de SLY.

Gramática Cool: Variables entre ángulos y tokens entre comillas. Además, se utilizan símbolos propios:

- * para representar 0 o más veces la expresión anterior
- + para representar 1 o más veces la expresión anterior
- | para representar la elección entre dos expresiones
- las expresiones entre corchetes son opcionales
- los paréntesis sirven para delimitar las expresiones

```

<Programa> ::= <Clase>+
<Clase> ::= "CLASS" "TYPEID" ["inherits" "TYPEID"] "{" (<Atributo> | <Metodo> )* "}" ";"
<Atributo> ::= "OBJECTID" ":" "TYPEID" ["ASSIGN" <Expresion>];"
<Metodo> ::= "OBJECTID" "(" (<Formal> ",")* <Formal> ")" ":" "TYPEID" "{" <Expresion> "}" ";"
| "OBJECTID" "(" (<Formal> ",")* <Formal> ")" ":" "TYPEID" "{" <Expresion> "}" ";"
<Formal> ::= "OBJECTID" ":" "TYPEID"
<Expresion> ::= "OBJECTID" "ASSIGN" <Expresion>
| <Expresion> "+" <Expresion>
| <Expresion> "-" <Expresion>
| <Expresion> "*" <Expresion>
| <Expresion> "/" <Expresion>
| <Expresion> "<" <Expresion>
| <Expresion> "<=" <Expresion>
| <Expresion> "=" <Expresion>
| "(" <Expresion> ")"
| "NOT" <Expresion>
| "ISVOID" <Expresion>
| "~" <Expresion>
| <Expresion> "@" "TYPEID" "." "OBJECTID" "(" " ")"
| <Expresion> "@" "TYPEID" "." "OBJECTID" "(" (<Expresion> ",")* <Expresion> ")"
| [ <Expresion> "." ] "OBJECTID" "(" (<Expresion> ",")* <Expresion> ")"
| [ <Expresion> "." ] "OBJECTID" "(" " ")"
| "IF" <Expresion> "THEN" <Expresion> "ELSE" <Expresion> "FI"
| "WHILE" <Expresion> "LOOP" <Expresion> "POOL"
| "CASE" <Expresion> "OF" ("OBJECTID" ":" "TYPEID" "DARROW" <Expresion>)+ ";" "ESAC"
| "NEW" "TYPEID"
| "{" (<Expresion> ";") + "}"
| "OBJECTID"
| "INT_CONST"
| "STR_CONST"
| "BOOL_CONST"

```

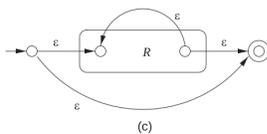


Figura 1: R*

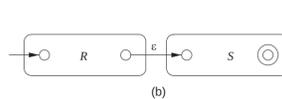


Figura 2: R·S

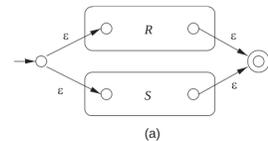


Figura 3: R|S

Inicializar: $\bar{Q} := Q$ y $\bar{q}_0 := q_0$.
para cada $p \in Q$ **encontrar** ε -*clausura*(p) **finaliza para**
 $\bar{F} := F \cup \{p : \varepsilon\text{-clausura}(p) \cap F \neq \emptyset\}$.
para cada $p \in Q$ **hacer**
si ε -*clausura*(p) $\neq \emptyset$, **entonces**
 $\bar{\delta}(p, a) := \bigcup_{q \in \varepsilon\text{-clausura}(p)} \varepsilon\text{-clausura}(\delta(q, a))$.
finaliza si, finaliza para cada