

Practica 1:
Desarrollo manual de un
analizador léxico para un lenguaje
sencillo

Youssef El Faqir El Rhazoui
Enrique Ávila Rodríguez

1. Clases léxicas del lenguaje

Para el lenguaje descrito en la práctica hemos identificado las siguientes clases léxicas. En la siguiente tabla se describe brevemente cada clase.

Clase Léxica	Descripción
num	Palabra reservada que indica números (enteros como flotantes)
bool	Palabra reservada para identificar valores booleanos (1, 0)
punt_coma	Identifica el final de una sentencia
ID	Referencia el nombre de una variable o cte.
sep_sec	Indica la separación entre la sección de declaración e instrucciones
asignacion	Referencia al operador de asignación
op_mas	Referencia al operador de suma binario o al unario de signo
op_menos	Referencia al operador de resta binario o al unario de signo
op_por	Referencia al operador de multiplicación
op_div	Referencia al operador de división
op_and	Referencia al operador conjuntor
op_or	Referencia al operador disyuntor
op_not	Hace referencia al operador lógico para la negación
op_menor_que	Hace referencia al operador de comparación '<'
op_mayor_que	Hace referencia al operador de comparación '>'
op_menor_igual	Hace referencia al operador de comparación '<='
op_mayor_igual	Hace referencia al operador de comparación '>='
op_igual	Hace referencia al operador de comparación '=='
op_distinto	Hace referencia al operador de comparación '!='
lit_num	Describe cómo va a ser un número en nuestro lenguaje, tanto entero como decimal
val_true	Palabra reservada para identificar el valor booleano True
val_false	Palabra reservada para identificar el valor booleano False
fin_arch	Identifica el final de fichero
abre_par	Identifica la apertura de un paréntesis
cierra_par	Identifica el cierre de un paréntesis

2. Especificación formal

A continuación, pasamos a especificar las clases léxicas mediante expresiones regulares. Lo que son caracteres o cadenas lo hemos denotado entre comillas simples.

Def. Aux	Clases Léxicas	Ignorables
letra -> [a-z, A-Z]	num -> 'num'	sep -> white_space EOL TAB
dig -> [0-9]	bool -> 'bool'	
p_ent -> (op_mas op_menos ε)(dig)+	punt_coma -> ';'	
p_dec -> '.'(dig)+	ID -> letra(letra dig '_')*	
p_exp -> ('e' 'E')p_ent	sep_sec -> '&&'	
	asignación -> '='	
	op_mas -> '+'	
	op_menos -> '-'	
	op_por -> '*'	
	op_div -> '/'	
	op_and -> 'and'	
	op_or -> 'or'	
	op_not -> 'not'	
	op_menor_que -> '<'	
	op_mayor_que -> '>'	
	op_menor_igual -> '<='	
	op_mayor_igual -> '>='	
	op_igual -> '=='	
	op_distinto -> '!='	
	lit_num -> p_ent (p_dec p_exp p_dec p_exp ε)	
	val_true -> 'true'	
	val_false -> 'false'	
	abre_par -> '('	
	cierra_par -> ')'	
	fin_arch -> EOF	

3. Diagrama de transiciones

Finalmente, en la ilustración 1 se muestra el autómata para la implementación. Las palabras reservadas se reconocen en una etapa posterior en la que se compara con un diccionario o un trie.

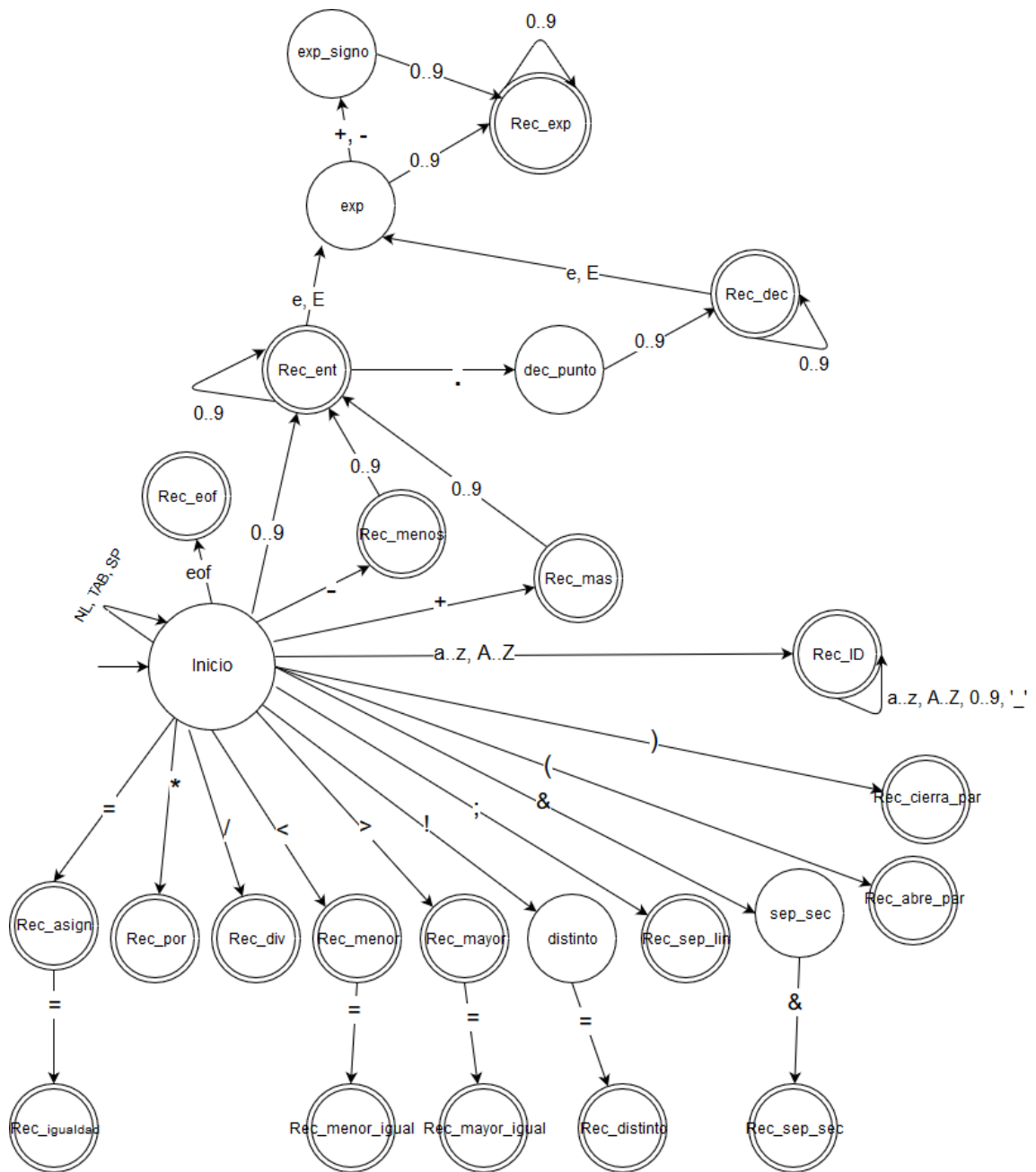


Ilustración 1: DFA