

UTN FRBA – SSL – Examen Final – 2017-12-11

Apellido, Nombre:		Legajo:		Nota:	
-------------------	--	---------	--	-------	--



- Resuelva el examen en tinta y en esta hoja; no se aceptan hojas adicionales.
- Durante el examen no se responde consultas; si lo necesita, escriba hipótesis de trabajo, las cuales también se evalúan.
- Para los ítems de *selección múltiple*, tilde (✓) sólo una opción, la mejor.

1. **Justifique** el valor de verdad de las siguientes afirmaciones:

- (1 punto) Durante su ejecución, el programa *lex* aplica el algoritmo de Thompson.
- (1 punto) El algoritmo de Thompson puede producir un AFN de solo un estado.

2. Sea

$$L = \{a^n b^n \mid n > 0\}$$

- (2 puntos) Formalice una gramática asociada, utilice solo un no terminal.
- (2 puntos) **Justifique** el valor de verdad de la siguiente afirmación:
La *regex* $(a\{1,3\}b\{1,3\})?$ representa un sublenguaje de L .

3. (1 punto) Analice **sintácticamente** la declaración

Racional `a, a(void), a[42], a([42]);`

Tilde la afirmación verdadera:

- ☐ `Racional` no es un tipo de dato válido.
- ☐ La declaración tiene un declarador incorrecto.
- ☐ Es ambigua por tener más de un árbol sintáctico.
- ☐ Es incorrecta por tener múltiples declaraciones de `a`.
- ☐ Las expresiones constantes no pueden usarse en declaraciones.

4. (3 puntos) Dada la declaración `char a[]="ABC";` analice **semánticamente** cada una de la siguientes expresiones, si es correcta, evalúela e indique su tipo de dato y valor. La primera expresión es un ejemplo:

Expresión	Tipo de dato	Valor
<code>a[0] < 'F'</code>	<code>int</code>	<code>1</code>
<code>a == &a[0]</code>		
<code>*a == a[0]</code>		
<code>&a[1] - &a[0]</code>		

1. Resolución

1.

- a. Verdadero. Lo utiliza para construir AFN a partir de las condiciones regex en cada regla lex.
- b. Falso. Para los dos casos básicos, $\{\epsilon\}$ y $\{a\}$, el algoritmo genera AFN con dos estados.

2.

a.

$$G = (\{S\}, \{a, b\}, \{(S, ab), (S, aSb)\}, S)$$

- b. Falso. Contiene palabras que no pertenecen a L, como por ejemplo, la palabra vacía y *aab*.

3.

✓ La declaración tiene un declarador incorrecto.

Justificación: En `a([42])`, luego de `(` no se espera `[`.

4.

Expresión	Tipo de dato	Valor
<code>a == &a[0]</code>	<code>int</code>	<code>1</code>
<code>*a == a[0]</code>	<code>int</code>	<code>1</code>
<code>&a[1] - &a[0]</code>	<code>int</code>	<code>1</code>

Para la diferencia, el tipo `ptrdiff_t` es también correcto.