

UTN FRBA – SSL – Examen Final – 2025-02-24

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



- Resuelva el examen en tinta y en esta hoja; no se aceptan documentos adicionales.
- Durante el examen no se responden consultas; si lo necesita, escriba hipótesis de trabajo, las cuales también se evalúan.

- Indique si las siguientes afirmaciones son verdaderas o falsas. Justifique su respuesta:
 - (1 punto) Los analizadores léxicos y sintáctico son los únicos encargados de detectar los errores en tiempo de compilación.
 - (1 punto) En C, el tipo de dato de una expresión es único.
 - (1 punto) Los AFN reconocen exactamente la misma clase de lenguajes que los AFD.
 - (1 punto) En C el tipo de asociatividad de todos los operadores está definida.
- (2 puntos) Las *keywords* son un sublenguaje del LF *identificadores*. ¿Es posible escribir una ER que represente el LF de los *identificadores* que no son *keywords*? ¿Por qué?
- Dadas las siguientes expresiones en C sintácticamente correctas:
 - (2 puntos) Escriba las declaraciones para que al evaluarse, cada expresión, produzcan un *valor-l no modificable* del tipo `double`:

Expresión	Declaración
<code>a.i</code>	
<code>*a.i</code>	
<code>(*a).i</code>	
<code>a->i</code>	

- (1 punto) Codifique una declaración que haga que la primera expresión (`a.i`) sea semánticamente incorrecta.
 - (1 punto) Codifique lo necesario para que al evaluar la segunda expresión (`*a.i`) se obtenga el `double` con valor 42.
- (Punto Extra) En la declaración `int *p[7], i()`; indique cuantos declaradores completos hay y cuantos declaradores. Justifique.

1. Una Resolución

1.
 - a. F. También lo hace el analizador semántico.
 - b. V. Se computa en tiempo de compilación en función a sus componentes, siguiendo las reglas semánticas de la especificación del lenguaje.
 - c. V. Tiene mismo poder de cómputo, reconocen los LR. Para toda ER se puede encontrar por lo menos un AFD o AFN; y para todo AFD y AFN se puede encontrar una ER. Para todo AFD existe un AFN y viceversa.
 - d. V. En la GIC.
2. Sí, es posible. Porque el complemento de un LR es R, y por lo tanto el complemento relativo con otro LR también lo es. También porque se puede construir un AF que trate como error cada keyword por una rama en particular.
3.
 - a.

Expresión	Declaración
<code>a.i</code>	<code>struct{const double i;} a;</code>
<code>*a.i</code>	<code>struct{const double *i;} a;</code>
<code>(*a).i</code>	<code>struct{const double i;} *a;</code>
<code>a->i</code>	<code>struct{const double i;} *a;</code>

- b. `int a;`
 - c. `double d=42;struct{double *i;}a={&d};`
4. Dos declaradores completos: `*p[7]` y `i()`. Y cinco declaradores: `*p[7]`, `p[7]`, `p`, `i()` e `i`.

v1.0.0-rc.2+2025-02-23