

UTN FRBA – SSL – Examen Final – 2018-02-26

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. (1 punto) Para llegar al AFD mínimo desde una ER es necesario aplicar exactamente tres algoritmos.

☐ Verdadero.

☐ Falso. Justificación si es falsa:

2. (1 punto) Dadas dos ER, el algoritmo del AFD mínimo es útil para determinar si representan el mismo LF.

☐ Verdadero.

☐ Falso. Justificación si es falsa:

3. (2 puntos) Dada la gramática

sentencia-if:

`if (expresión) sentencia`

`if (expresión) sentencia else sentencia`

Indique si la gramática es $LL(1)$. Si lo es justifique, si no, reescríbala para que sí sea:

4. Dado el PAS, resuelva los siguientes ítems:

```
void S(){
    switch(GetProximoToken()){
        case A: Match(A); S(); Match(C); return;
        case B: Match(B); return;
        default: // TODO
    }
}
```

a. (1 punto) El PAS es sintácticamente correcto.

☐ Verdadero.

☐ Falso. Justificación si es falsa:

b. (1 punto) Dibuje el árbol de expresión de la expresión de la sentencia etiquetada con `case B`:

c. (1 punto) La sentencia etiquetada con `default` no fue desarrollada por el programador original, por eso tiene el comentario `TODO (para hacer)`, complétela arriba, en el propio código.

d. (1 punto) Defina el conjunto de las producciones de la gramática asociada al PAS:

e. (1 punto) Escriba una *regex* que represente el Σ asociado al PAS:

f. (1 puntos) Calcule $\text{Primero}(S)$:

1. Resolución

1.

✓ Verdadero.

2.

✓ Verdadero.

3. *sentencia-if*:

if (expresión) sentencia else-opcional

else-opcional:

ε

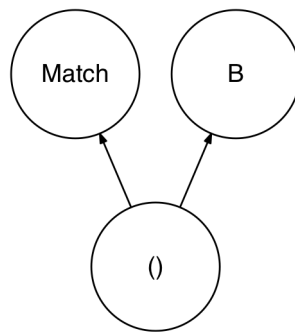
else sentencia

4.

a.

✓ Falso. No se espera el token `}` después de `default:`, se espera una *sentencia*.

b.



c. `InformarErrorSintáctico()`;

d.

$\{S \rightarrow ASC, S \rightarrow B\}$

e. `[A-C]`

f.

$\text{Primero}(S) = \{A, B\}$