

UTN FRBA – SSL – Examen Final – 2021-09-24

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



- Resuelva el examen en el documento compartido para edición; no se aceptan documentos adicionales.
- Durante el examen no se responden consultas; si lo necesita, escriba hipótesis de trabajo, las cuales también se evalúan.

1. (2 puntos) Indique y justifique cuál es la operación que se realiza más frecuentemente sobre una tabla de símbolos.
2. (2 puntos) Escriba el prototipo en C de la función transición de un AFD. Justifique.
3. Dado el siguiente fragmento: $a < b < c$
 - a. (1 punto) Indique la secuencia de caracteres devueltos por `ungetc` durante el análisis léxico.
 - b. (1 punto) Indique la asociatividad mediante paréntesis redundantes.
 - c. Para cada situación pedida a continuación escriba **una** declaración que haga...
 - i. (1 punto) ... semánticamente **inválido** al fragmento.
 - ii. (1 punto) ... **cero** al valor resultante sin usar el tipo `int`.
 - iii. (1 punto) ... **uno** al valor resultante sin usar el tipo `int`.
 - d. (1 punto) Escriba una expresión que evalúe si un valor b está dentro del intervalo abierto (a, c) . Agregue la declaración que haga semánticamente válida a esa expresión, no es necesario inicializar.
 - e. (Punto extra) Presente un contexto semántico donde la anterior expresión, aunque semánticamente correcta, no pueda ser utilizada. Justifique.

1. Una Resolución

1. *Búsqueda*. Ante cada identificador se busca en la tabla. Las inserciones y remociones son menos frecuentes comparativamente. Para insertar es necesario primero determinar si ya existe, para remover es necesario primero buscarlo.

2. Un par de alternativas:

Las dos reciben el caracter leído y el estado actual, y retornan el siguiente estado.

- `int T(int, int);`
- `Estado T(Estado, char); // typedef ... Estado;`

3. a. `<b<c`

b. `(a<b)<c`

c. i. `struct {double x,y;} a,b,c;`

ii. `char a=1,b=2,c=0;` entonces `a<b<c` vale 0

iii. `char a='a',b='b',c='c';` entonces `a<b<c` vale 1

d. `a<b && b<c`

e. `(a<b && b<c)=42`

La expresión `(a<b && b<c)` es semánticamente válida, pero en este contexto, a la izquierda de una asignación, se espera un valor-l modificable.

v1.1.0 2021-09-25