

## UTN FRBA – SSL – Examen Final – 2021-03-10

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.

Dada la siguiente definición:

```
int f(int x, int y){  
    int z=1;  
    for(int i=1; i<=y; ++i)  
        z*=x;  
    return z;  
}
```

1. Analice Léxicamente:

- (1 punto) Indique la cantidad de tokens de la tercera expresión de la sentencia for.
- (1 Punto) Indique la cantidad de veces que el símbolo = es un lexema. Justifique.

2. Analice Sintácticamente:

- (1 punto) Indique si faltan llaves para que f sea sintácticamente correcta. Justifique.
- (1 punto) Indique la cantidad de declaraciones **en la sentencia compuesta**.
- (1 punto) Indique la cantidad de expresiones completas **en la definición**.

3. Analice Semánticamente:

- (1 punto) Indique si la tercera expresión del for debiera ser i++. Justifique.
- (1 punto) Considerando como único contexto la definición dada, escriba una declaración que use f y sea semánticamente correcta.
- (1 punto) Indique el alcance de f.
- (1 punto) Indique el alcance de z.
- (1 punto) Indique los valores de x e y que hacen que la función retorne.

4. (Punto extra) Analice la pragmática de la función y luego escriba un nuevo prototipo para f que sea superador, es decir que sea mejor según sus propios criterios. Puede cambiar el identificador de la función y los tipos de datos. Justifique.

## 1. Una Resolución

1.
  - a. Dos: ++ e i.
  - b. Dos: En ambas inicializaciones.
2.
  - a. La sintaxis de la sentencia `for` no necesita llaves, su última parte es una sentencia que puede ser compuesta, y en ese caso, por la sintaxis de las sentencias compuestas, sí tendría llaves, pero no por el `for`.
  - b. Dos: Declaraciones de `z` e `i`.
  - c. Seis: Inicialización de `z`, de `i`, segunda y tercera expresión del `for`, sentencia expresión, valor de retorno.
3.
  - a. Las expresiones evalúan diferente, pero el efecto de lado es el mismo, así que no es **necesario** cambiarla.
  - b. `int r=f(1,2);`
  - c. Desde dónde comienza su definición hasta el fin de la unidad de traducción, sin contar ocultamientos.
  - d. La función `f`.
  - e. Para todo valor de `x` y de `y` la función retorna.

1. `long Potencia(int, unsigned);`

v1.0.0-beta.1 2021-03-13