

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN
LENGUAJES DE PROGRAMACIÓN
TERCERA EVALUACIÓN - I TÉRMINO 2016-2017

Nombre: _____ Matrícula: _____ Paralelo: _____

TEMA 1. CONCEPTOS GENERALES (18 puntos)

1. Encierre el literal de la respuesta correcta. (3 puntos)

¿Qué es un lenguaje de programación?

- a) Es un lenguaje formal diseñado para realizar procesos que pueden ser llevados a cabo por máquinas como las computadoras.
- b) Es un software que permite definir las palabras reservadas, tipos de datos y reglas sobre los que van a trabajar los programadores.
- c) Es un compilador estable y potente para la solución de problemas humanos y los entienda la PC.
- d) Se utilizan como una alternativa diferente de los compiladores para traducir lenguajes de alto nivel.

Las evaluaciones de funciones LISP deben devolver:

- a) un único valor sin producir modificaciones en los parámetros recibidos.
- b) varios valores con modificaciones en los parámetros recibidos.
- c) un valor nulo sin producir modificaciones en los parámetros recibidos.
- d) varios valores sin modificaciones en los parámetros recibidos.

El tipado fuerte es una característica de los lenguajes de programación para:

- a) especificar las restricciones que deben aplicarse sobre las operaciones de los diferentes tipos de datos.
- b) evitar errores en tiempo de ejecución.
- c) detectar errores sintácticos y semánticos.
- d) Todas las anteriores.

2. Conteste F o V según corresponda y justifique su respuesta. (2 puntos)

- a) Un intérprete se diferencia de un compilador ya que no realiza los 3 análisis: léxico, sintáctico, semántico. ()
- b) El uso de parámetros predeterminados evita el error de tipos y facilita la recursividad. ()

a.

b.

3. Mencione la diferencia entre un error sintáctico y un error semántico. Coloque un ejemplo de error sintáctico en Prolog. (3 puntos)

4. Mencione dos funciones que un compilador define a través de reglas semánticas. (2 puntos)

5. Complete la siguiente tabla comparativa de Lenguajes de Programación. (8 pts)

Lenguaje	Paradigma	Tip. fuerte	Característica importante
Python			
LISP			
SML			
Prolog			

TEMA 2. EXPRESIONES REGULARES + RUBY (20 pts)

6. Diseñe una función en Ruby que permita controlar el vocabulario utilizado en los foros del SIDWEB. Suponiendo que la palabra "molestar" se consideraría un verbo de insulto, diseñe una expresión regular que detecte esta palabra indebida a pesar de que no se escriba correctamente. Al menos valide para los siguientes ejemplos: molest@r, m0lest@r, m*lest@r, mo1est@r, mol3star.

TEMA 3. ÁRBOL SINTÁCTICO + LISP (30 pts)

7. Diseñe un programa en LISP que permita calcular la serie fibonacci, luego grafique su Parser tree. Por ejemplo, si ingresa el 8 su impresión debería ser: 34
Serie Fibonacci: 1, 2, 3, 5, 8, 13, 21, 34, 55...
Programa (15 pts)

Parser Tree (15 ptos)

TEMA 4. PROGRAMACIÓN FUNCIONAL - SML (15 ptos)

8. Considere la siguiente función en SML, sabiendo que:

- **String.sub**(cadena, pos) obtiene un carácter de la cadena de texto de acuerdo a una posición enviada.
- **String.substring**(cadena, inicio, fin) obtiene una subcadena de texto de acuerdo a un inicio y fin.
- **size** se utiliza para obtener la longitud de una cadena de texto.

Analice el código, realice una prueba de escritorio, y escriba ¿cuál sería su resultado si $s = \text{"101010"}$.?

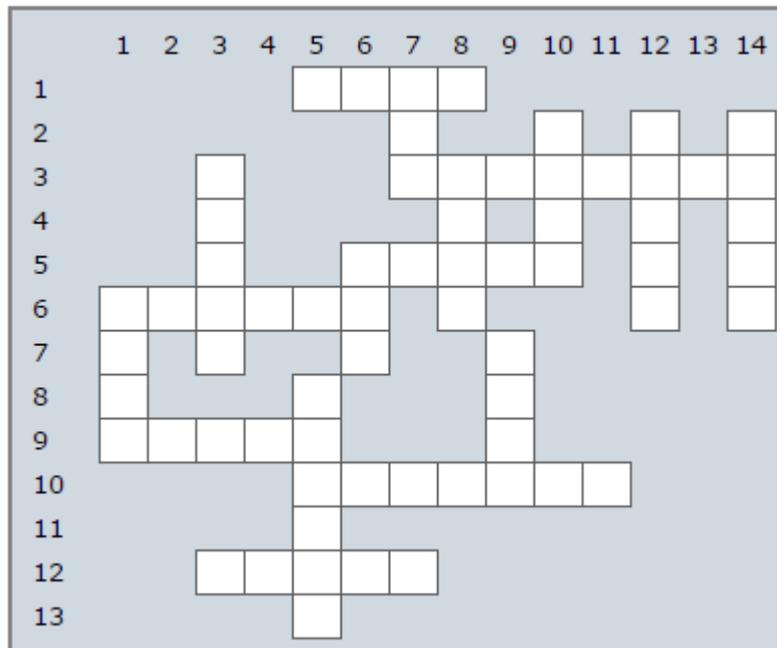
```
fun funcion s =  
  if size s <= 1 then true  
  else if String.sub (s,0) = String.sub(s, size s - 1)  
  then  
    funcion (String.substring(s, 1, size s - 2))  
  else false;
```

Prueba de Escritorio (10 ptos)

Resultado (5 ptos).

TEMA 5. TIPOS DE DATOS (17 ptos)

9. Resuelva el siguiente crucigrama:



Horizontal

- 1: En Ruby permite definir la siguiente estructura: { :leia => "Princess from Alderaan", :han => "Rebel without a cause", :luke => "Farmboy turned Jedi" }
- 3: Almacena números como 1/2, 3/4, 8/2, es decir, números con un valor de numerador y un denominador.
- 5: En C#, Java o Scala este tipo de dato permite almacenar valores enteros de 16 bits
- 6: Tipo de punto flotante de precisión doble
- 9: Permite definir la siguiente estructura (3, "a", 9.0)
- 10: Tipo de dato que permite en Python definir $a = 2.1 + 7.8j$
- 12: En ruby permite definir varios elementos contiguos: ["hola", "esto", "es", "una", "prueba"]

Abajo

- 1: Tipo de dato que permite definir diccionarios en Python
- 3: Palabra reservada de LISP para definir una función.
- 5: Tipo de dato en SML que permite definir lo siguiente $vec = \{ x : real, y : real \}$;
- 6: En Python permite definir conjuntos
- 7: Tipo de dato en Python para almacenar cadenas
- 8: En LISP se refiere a un número, cadena, símbolo, etc.
- 9: En Python permite definir valores True o False.
- 10: En Python este tipo de dato permite definir algo como esto: ["Ecuador", "Perú", "Colombia"]
- 12: Es un tipo de dato derivado, como una estructura, con miembros que comparten el mismo espacio de almacenamiento. No permite la comprobación de tipos de sus elementos.
- 14: Permite definir el siguiente valor: 0.348