

**LENGUAJES DE PROGRAMACIÓN  
PRIMERA EVALUACIÓN II TÉRMINO 2017-2018**

Nombre: \_\_\_\_\_

Fecha: \_\_\_\_\_

**Indicaciones:**

- Leer con detenimiento las preguntas planteadas.
- Poner en práctica sus principios éticos.
- El tiempo para el desarrollo de la evaluación es de 2 horas.

**1) De los siguientes lenguajes de programación, cuáles NO corresponden a lenguajes orientados a objetos y/o estructurados. (5 pt)**

- |                    |             |                           |                       |                       |
|--------------------|-------------|---------------------------|-----------------------|-----------------------|
| a) Ruby, Smalltalk | b) Lisp, ML | c) Python,<br>Objective-C | d) Prolog,<br>Haskell | e) Smalltalk,<br>Java |
|--------------------|-------------|---------------------------|-----------------------|-----------------------|

**2) En cierto lenguaje de programación es posible definir punteros a cualquier tipo específico definido en el lenguaje, excepto a arreglos. Por tal razón, muchas estructuras de datos potencialmente útiles para el usuario no pueden ser definidos. Esto refleja un problema de: (5 pt)**

- |                        |                |                 |                             |                  |
|------------------------|----------------|-----------------|-----------------------------|------------------|
| a) Chequeo<br>de tipos | b) Abstracción | c) Optimización | d) Manejo de<br>excepciones | e) Ortogonalidad |
|------------------------|----------------|-----------------|-----------------------------|------------------|

**3) Escriba tres conceptos básicos de la programación orientada a objetos. (9 pt)**

**4) ¿En qué contribuye la forma Backus-Naur a las gramáticas libres de contexto?. (6 pt)**

- a) La forma Backus-Naur (BNF) es una técnica que contribuye con la medición del tiempo de ejecución en programas interpretados.
- b) La forma Backus-Naur es una técnica de notación para gramáticas libres de contexto, generalmente utilizada para describir sintaxis en lenguajes de programación.
- c) La forma Backus-Naur contribuyó con la programación orientada a objetos en la introducción de los conceptos relacionados con encapsulamiento.
- d) La forma Backus-Naur (BNF) contribuye con una notación formal para gramáticas libres de contexto, generalmente utilizada para describir semántica en lenguajes de programación.
- e) La forma Backus-Naur (BNF) contribuye a las gramáticas libres de contexto en la detección de ambigüedades.

5) Dada la siguiente gramática, ¿cuál de las siguientes declaraciones permitiría demostrar su ambigüedad? (15pt)

```
<start> → <term>
<term> → <id> | <term> + <term>
<id> → a | b | c
```

- a) a+b      |      b) a      |      c) c+a      |      d) c      |      e) a+b+c

Justifique:

6) Considerando la gramática dada, ¿cuáles de las siguientes declaraciones son generadas por esta? (15 pt)

```
<I> → x <I> z <Y> | <X> | y
<X> → z <X> | z
<Y> → w | <X>
```

- a) xyzw      |      b) xzzyw      |      c) xzzzyzz      |      d) xzw      |      e) xzzz

7) Escriba en clojure la siguiente función en Python (20 pt)

<pre>def factorial(n):     if n == 0:         return 1     else:         return n * factorial(n-1)</pre>	
--	--

8) Convierta la expresión regular  $pb^*|b^+$  en un autómata de estado finito determinista usando el método de conversión NFA-DFA. (25pts)