

Universidad Nacional Autónoma de México  
Primer Examen Parcial –Parte B–  
Lenguajes de Programación  
Semestre 2023-2  
7 de marzo de 2023

Nombre: \_\_\_\_\_ Calificación: \_\_\_\_\_ Puntos extra: \_\_\_\_\_

1. (1 pt.) Explica qué es un *idiom*. Da un ejemplo en el ámbito de la programación.
2. (2 pts.) Explica los 4 paradigmas de lenguajes de programación y da al menos 1 ejemplo de algún lenguaje de programación que pertenezca a cada uno de los paradigmas.
3. (1 pt.) ¿Cuál es la complejidad del algoritmo de sustitución? Justifica tu respuesta usando algún ejemplo y explicándolo lo más detalladamente posible.
4. (1 pt.) Da una expresión usando la gramática FWAE tal que un mismo nombre de identificador (supongamos  $x$ ) aparezca en la misma expresión como identificador de ligado dos veces (con el mismo nombre de identificador), aparezca ligado al menos una vez y aparezca exactamente una única vez como identificador libre. Especifica a qué tipo de identificadores te refieres (de ligado, ligadas y libre) en tu expresión.
5. (1 pt.) Convierte el siguiente código usando índices de Bruijn o direcciones léxicas.

```
{with {x -1}
  {with {y 1}
    {with {z 0}
      {with {w {+ 1 2}}
        {+ x {+ y {+ {+ z w} {+ w z}}}}}}}}
```

6. (1 pt.) Convierte el siguiente código con índices de Bruijn a código dentro de la gramática WAE. Las instancias de ligado se deben de nombrar como “x”, “y”, “z”, “w”, “v”, con respecto al orden de aparición de las mismas.

```
{with 1
  {with 2
    {with 3
      {with {+ <:1 0> <: 0 0>}
        {with 4
          {+ <: 0 0> {+ <:1 0> {+ <:2 0> {+ <:3 0> <:4 0>}}}}}}}}
```

7. (1 pt.) A qué se le conoce como azúcar sintáctica en un lenguaje de programación.
8. (4 pts.) Ponga el ambiente en forma de pila (stack) para la siguiente expresión, y evalúe la siguiente expresión usando
  - a) Alcance estático.
  - b) Alcance dinámico.

es necesario especificar cada una de las expresiones a evaluar con los respectivos valores.

```

{with {a -1}
  {with {b 1}
    {with {a 1}
      {with {foo1 {fun {x} {* x {+ a b}}}}
        {with {b -1}
          {with {a -1}
            {foo1 7}}}}}}}}

```

9. (1 pt.) (a) Explica en un renglón, qué hace la siguiente función recursiva (no tienes que explicar línea por línea sino en general qué hace, (b) usa al menos 3 ejemplos para que primero verifiques qué hace la función (haz ejecuciones, al menos 3), y (c) dale un nombre mnemotécnico a ésta:

```

(define (secreto l n)
  (cond
    [(empty? l) empty]
    [(zero? n) l]
    [else (secreto (cdr l) (sub1 n))]))

```