Universidad del Valle

Segundo examen opcional - parte práctica
Fundamentos de lenguajes de programación
Duración: 1.5 horas

Carlos Andres Delgado S, Ing *

11 de Junio de 2018

Nombre:_____ **Código:**_____

Importante: La única página autorizada para abrir es el campusvirtual, si abre cualquier otra o si se le ve chateando será considerado fraude y se anulará el opcional. Debe enviar los interpretadores solicitados con comentarios explicando que realizó en el enlace provisto en el campus antes de que cierre.

1. Tipos [30 puntos]

Todo este proceso realizado en el interpretador de chequeo de tipos.

1.1. Cadenas de texto (15 puntos)

Agregar la especificación de los textos entre comillas y la primitiva concatenar &, ejemplo:

```
let
    a = "papa"
    b = "ymama"
    in
        let
            c = &(a,b)
            in
                c
```

Debe arrojar "papamymama". La regla de & es:**(string*string)− >string**. Explique todo el proceso que realizó para implementar este tipo. Muestre 2 ejemplos con error de tipo.

1.2. Listas(15 puntos)

Agregar la especificación listas, estas van entre [] y los elementos separados por comas, ejemplo:

```
a = [1,2,3,4,5]
```

La regla de las listas es: [t1,t2,t3,t4...,tn] donde se debe cumplir t1=t2=t3=t4=...=tn, por lo que es correcto tener una expresión

```
a = [1,2,3,4,5]
```

Pero da error de tipos una expresión

```
a = [1,proc(x) x,4,4,5]
```

Ya que todos los elementos de la lista deben tener el mismo tipo. Muestre dos ejemplos con error de tipo.

*carlos.andres.delgado@correounivalle.edu.co

2. Objetos (20 puntos)

En el interpretador de **objetos planos**.
Permita la sobrecarga. La firma de un método (su identidad) es normalmente su nombre, sin embargo, en sobrecarga su identidad es su nombre más el número de argumentos que recibe, ejemplo:

```
1 method a() 1
2 method a(x,y) +(x,y)
```

La firma del primer método es **a0** y la del segundo es **a2**, supongamos que o1 es un objeto de la clase, entonces los métodos los invocamos de la siguiente forma:

```
1 send o2 a() % Retorna 1
2 send o2 a(1,2) % Retorna 3
```

Incluya un ejemplo de un objeto con al menos herencia de 2 clases y 3 métodos sobrecargados incluyendo initializate.

¡Éxitos!