Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Escuela de Ciencias Exactas y Naturales
Departamento de Ciencias de la Computación
Análisis de Lenguajes de Programación

Nombre y Apellido: Legajo:

Examen Final

1. Un lenguaje, según lo visto en la materia, puede ser definido por su sintáxis, su semántica y un sistema de tipos. A continuación se presenta un lenguaje, definiendo estos componentes.

Sintaxis:

$$
\begin{array}{rl}
t & ::= x \mid \mathsf{suc}\ t \mid 0 \mid \mathsf{unit} \mid \mathsf{inl}\ t \mid \mathsf{inr}\ t \mid \mathsf{case}\ t\ \mathsf{of}\ \mathsf{inl}\ x \Rightarrow t; \mathsf{inr}\ x \Rightarrow t \\
v & ::= nv \mid \mathsf{unit} \mid \mathsf{inl}\ v \mid \mathsf{inr}\ v \\
nv & ::= 0 \mid \mathsf{suc}\ nv \\
T & ::= \mathbb{N} \mid \mathsf{B} \mid T + T
\end{array}
$$

Semántica:

$$\frac{t \to t'}{\mathsf{suc}\ t \to \mathsf{suc}\ t'} \quad \text{(E-Suc)}$$

Sistema de tipos:

$$\Gamma \vdash \mathsf{unit} : \mathsf{B} \quad \text{(T-Unit)}$$

$$\frac{\Gamma \vdash t : \mathbb{N}}{\Gamma \vdash \mathsf{suc}\ t : \mathbb{N}} \quad \text{(T-Suc)}$$

$$\frac{\Gamma \vdash t : T}{\Gamma \vdash \mathsf{inl}\ t : T + T'} \quad \text{(T-Inl)}$$

$$\frac{\Gamma \vdash t : T'}{\Gamma \vdash \mathsf{inr}\ t : T + T'} \quad \text{(T-Inr)}$$

$$\frac{\Gamma \vdash t : T_1 + T_2 \qquad \Gamma, x_1 : T_1 \vdash t_1 : T \qquad \Gamma, x_2 : T_2 \vdash t_2 : T}{\mathsf{case}\ t\ \mathsf{of}\ \mathsf{inl}\ x_1 \Rightarrow t_1; \mathsf{inr}\ x_2 \Rightarrow t_2 : T} \quad \text{(T-Case)}$$

$$\frac{t \to t'}{\mathsf{inr}\ t \to \mathsf{inr}\ t'} \quad \text{(E-Inr)}$$

$$\frac{t \to t'}{\mathsf{inl}\ t \to \mathsf{inl}\ t'} \quad \text{(E-Inl)}$$

$$\mathsf{case}\ \mathsf{inl}\ v\ \mathsf{of}\ \mathsf{inl}\ x_1 \Rightarrow t_1; \mathsf{inr}\ x_2 \Rightarrow t_2 \to t_2\ [x_2\ /\ v] \quad \text{(E-Case1)}$$

$$\mathsf{case}\ \mathsf{inr}\ v\ \mathsf{of}\ \mathsf{inl}\ x_1 \Rightarrow t_1; \mathsf{inr}\ x_2 \Rightarrow t_2 \to t_1\ [x_1\ /\ v] \quad \text{(E-Case2)}$$

a) Enunciar las propiedades de seguridad vistas en clase.

b) El lenguaje dado no cumple la propiedad de progreso. Dar un ejemplo de por qué no cumple esta propiedad y modificar el lenguaje para que sí la cumpla.

c) ¿De qué manera podría definirse el tipo Bool usando la suma de tipos? Dar definiciones de true, false y **if** como formas derivadas usando sumas de expresiones.

2. Se representarán los tipos de un lenguaje con el siguiente tipo de datos:

```
data Type v = TVar v                    -- Variable de tipo
            | TInt                      -- Tipo Integer
            | Fun (Type v) (Type v)     -- Tipo de función
```

a) Crear la instancia de la clase *Monad* para Type (cuyo efecto sea la sustitución de variables) y probar que es una mónada.

b) Definir las siguientes funciones:

i. apply :: $(a \to \mathsf{Type}\ b) \to \mathsf{Type}\ a \to \mathsf{Type}\ b$, que aplique una sustitución a un tipo.

ii. comp :: $(a \to \mathsf{Type}\ b) \to (c \to \mathsf{Type}\ a) \to (c \to \mathsf{Type}\ b)$, que permita componer sustituciones.

3. Usando la representación de listas en lambda cálculo dada en clase, definir las funciones *isNil*, *head* y *tail*.