PLP - Segundo Parcial - 2$^{\text{do}}$ cuatrimestre de 2018

Este examen se aprueba obteniendo al menos dos ejercicios bien menos (`B-`) y uno regular (`R`), y se promociona con tres ejercicios bien menos (`B-`) y las respuestas adecuadas a las preguntas teóricas. Las notas para cada ejercicio son: `-`, `I`, `R`, `B-`, `B`. Poner nombre, apellido, número de orden y cantidad de hojas en la primera hoja, y numerar las hojas. Se puede utilizar todo lo definido en las prácticas y todo lo que se dio en clase, colocando referencias claras.

Ejercicio 1 - Subtipado

Se extiene el Cálculo Lambda para soportar **conjuntos** y **multiconjuntos**.

$\sigma ::= \cdots \mid \mathsf{Set}_\sigma \mid \mathsf{MultiSet}_\sigma$

$M ::= \cdots \mid M \in N \mid \#(M, N) \mid M \setminus \{N\}$

$$\frac{\Gamma \triangleright M : \sigma \qquad \Gamma \triangleright N : \mathsf{Set}_\sigma}{\Gamma \triangleright M \in N : \mathsf{Bool}}(\text{T-PERTENECE}) \qquad \frac{\Gamma \triangleright M : \sigma \qquad \Gamma \triangleright N : \mathsf{MultiSet}_\sigma}{\Gamma \triangleright \#(M, N) : \mathsf{Nat}}(\text{T-APARICIONES})$$

$$\frac{\Gamma \triangleright M : \mathsf{Set}_\sigma \qquad \Gamma \triangleright N : \sigma}{\Gamma \triangleright M \setminus \{N\} : \mathsf{Set}_\sigma}(\text{T-SACARUNO}) \qquad \frac{\Gamma \triangleright M : \mathsf{MultiSet}_\sigma \qquad \Gamma \triangleright N : \sigma}{\Gamma \triangleright M \setminus \{N\} : \mathsf{MultiSet}_\sigma}(\text{T-SACARUNOM})$$

Donde:

- $M \in N$ que indica si el elemento M se encuentra en el conjunto N
- $\#(M, N)$ retorna la cantidad de apariciones del elemento M en N
- $M \setminus \{N\}$ que elimina un elemento de un **conjunto** o **multiconjunto**.

a. Proponer las reglas de subtipado para los nuevos tipos justificando cuando un tipo es reemplazable por otro. Se pide analizar la covarianza, contravarianza e invarianza.

b. Se agrega el siguiente término $M ::= \cdots \mid M + N$ cuya regla de tipado es:

$$\frac{\Gamma \triangleright M : \mathsf{MultiSet}_\sigma \qquad \Gamma \triangleright N : \sigma}{\Gamma \triangleright M + N : \mathsf{Unit}}(\text{T-AGREGARUNO})$$

Justificar si hay o no impacto en las reglas de subtipado previamente propuestas.

Ejercicio 2 - Objetos

a. Para el siguiente fragmento de código Javascript describir que se imprime en consola justificando cada respuesta:

```
1. let o1 = {x: 10, y: function(){this.x = this.x*3; return this.x}};
2. let o2 = {x: 20};
3. o2.__proto__.z = 30;
4. o2.y = o1.y;
5. o2.y();
console.log(o2.x);
console.log(o1.x);
console.log(o2.z);
console.log(o1.z);
function G(){};
G.prototype.z = 40;
let o3 = new G();
console.log(o3.x);
console.log(o3.z);
}
```

b. Definir en el cálculo de objetos, el objeto `Vacio` que representa el conjunto vacío y sabe responder los siguientes mensajes:

- hayElementos(), que devuelve verdadero si el conjunto contiene elementos.
- agregar(x), que devuelve el objeto que agrega x al conjunto.
- sacar(x), que devuelve el objeto que saca x del conjunto.
- pertenece(x), que indica si x pertenece al conjunto.

Ejemplos:

`Vacio.agregar(2).sacar(2).hayElementos()` $\rightarrow$ `false;`

`Vacio.agregar(2).agregar(3).sacar(2).pertenece(3)` $\rightarrow$ `true;`

Ejercicio 3 - Lógico

Implementar los predicados respetando en cada caso la instanciación pedida. Los generadores deben cubrir todas las instancias válidas de aquello que generan sin repetir dos veces la misma. No usar cut (`!`) ni predicados de alto orden como `setof`, con la única excepción de `not`.

Sean A y B dos conjuntos. Una relación binaria R va a estar representada como una lista de pares (a, b) con $a \in A$ y $b \in B$. Se pide implementar los siguientes predicados:

a. `relacion(+A,+B,-R)` que es verdadero cuando `R` es una relación binaria sobre los conjuntos `A` y `B`. Recordar que dados dos conjuntos A y B, una relación definida de A en B es cualquier subconjunto del producto cartesiano $A \times B$.

b. `funcion(+D,+I,-F)` que es verdadero cuando `F` es una función, es decir, es una relación a la cual se anade la condición de que a cada valor del dominio le corresponde uno y sólo un valor de la imagen.

Preguntas Teóricas

a. Decidir si por verdadero o falso cada una de las siguientes afirmaciones, justificando cada una de sus respuestas:

   I) No existe resolvente para $\{P(g(a,x), x, f(a))\}$ y $\{\neg(P(x,y,y)\}$ porque los literales no unifican;

   II) $\forall x.\forall y.(P(x, f(x)) \wedge \neg P(y, f(y)))$ es una forma normal de Skolem de $\forall x.\forall y.\exists z.(P(x,z) \wedge \neg P(y,z))$

b. Definir en el cálculo de objetos, la clase `Conjunto` cuyas instancias responden los mensajes descriptos en el item 2.b.

c. Sea un programa en Prolog que redefine el **not** como:

```
not(G) :- call(G), fail, !.
not(G).
```

   Comparar el resultado de evaluar `not(P)` con la definición original de `not`.