

Segundo parcial

NOTA: este parcial es a libro abierto. Se permite tener cualquier material manuscrito o impreso, pero no se permite el uso de dispositivos electrónicos. El parcial dura 3 horas y se califica con una nota numérica de 1 a 10. Se requiere ≥ 4 en ambos parciales para aprobar la materia. Para promocionar se requiere nota ≥ 6 en ambos parciales y promedio ≥ 7 .

Ejercicio 1. Extender la siguiente gramática $G = (\{E, O\}, \{y, \text{neg}, \text{verdadero}, \text{falso}, \text{var}, (,)\}, P, E)$ a una gramática con un atributo **ast** que sintetice el AST de una expresión booleana **haciendo constant-folding**. Las producciones son:

$$E \rightarrow E \text{ y } E \mid \text{neg } E \mid \text{verdadero} \mid \text{falso} \mid \text{var} \mid (E)$$

El operador binario “y” representa la conjunción lógica, “neg” la negación, “var” representa una variable y “verdadero” y “falso” son constantes. Suponemos que las variables **var** vienen acompañadas de un atributo nombre de tipo **String**. El AST está dado por el siguiente tipo, con la interpretación esperable:

```
data Expr = ExprAnd Expr Expr | ExprNot Expr | ExprTrue | ExprFalse | ExprVar String
```

Importante: se debe hacer *constant-folding*. Es decir, para todas las subexpresiones que no involucren variables, el AST sintetizado debe ser un árbol de un único nodo que representa un valor constante.

Ejercicio 2. Dado el siguiente programa:

```

1 | t1 := 10          7 | impar:
2 | start:          8 | t1 := 3 * t1
3 | t2 := t1 mod 2  9 | t1 := t1 + t2
4 | jumpIf= t2 1 impar 10 | jumpIf< t1 t2 start
5 | t1 := t1 div 2 11 | t1 := 2 * t1
6 | t2 := -1
    
```

- a. Construir el grafo de flujo de control.
- b. Calcular las definiciones de alcance para **todas** las líneas del programa. Indicar los conjuntos *gen*, *kill*, *in* y *out* para cada nodo del grafo de flujo.

Ejercicio 3. Dado el siguiente programa:

```

0 | t0 := 0
1 | t1 := 1
2 | t2 := t0 + t1
3 | t3 := t1 + t2
4 | t4 := t2 + t3
5 | t5 := t3 + t4
6 | t6 := t4 + t5
    
```

- a. Construir el grafo de interferencia G .
- b. Hallar el mínimo k tal que G es k -coloreable.
- c. Reescribir el programa usando solamente k registros.

Ejercicio 4. Considerar el lenguaje con los siguientes términos y tipos:

$$M ::= \text{id} \mid (M \circ M) \mid \langle M, M \rangle \mid \text{fst} \mid \text{snd} \qquad \alpha ::= \mathbf{Bool} \mid \mathbf{Int} \mid (\alpha \times \alpha)$$

Los juicios de tipado son de la forma $M : \alpha \rightarrow \beta$ y están dados por las siguientes reglas:

$$\begin{array}{c}
 \frac{}{\text{id} : \alpha \rightarrow \alpha} \text{T-ID} \qquad \frac{M : \alpha \rightarrow \beta \quad N : \beta \rightarrow \gamma}{(N \circ M) : \alpha \rightarrow \gamma} \text{T-COMPOSE} \\
 \\
 \frac{M : \alpha \rightarrow \beta \quad N : \alpha \rightarrow \gamma}{\langle M, N \rangle : \alpha \rightarrow (\beta \times \gamma)} \text{T-PAIR} \qquad \frac{}{\text{fst} : (\alpha \times \beta) \rightarrow \alpha} \text{T-FST} \qquad \frac{}{\text{snd} : (\alpha \times \beta) \rightarrow \beta} \text{T-SND}
 \end{array}$$

a. Proponer un término M apropiado para que se pueda derivar el siguiente juicio y exhibir la derivación:

$$M : \alpha \rightarrow (\alpha \times \alpha)$$

b. Proponer tipos τ_1 y τ_2 apropiados para que se pueda derivar el siguiente juicio y exhibir la derivación:

$$\langle (\text{snd} \circ \text{snd}), \langle (\text{fst} \circ \text{snd}), \text{fst} \rangle \rangle : \tau_1 \rightarrow \tau_2$$

Justificar todas las respuestas.