

EJERCICIOS sobre SEMANTICA DENOTACIONAL

1) Da el significado, según la semántica denotacional, del siguiente programa PAM:

```

read n; p:= 1; i := n ;
while i > 0 do p := p * x ; i := i - 1 end;
write p

```

2) Describe en semántica denotacional las instrucciones **for**, **loop** y **case** vistas en la hoja de ejercicios de semántica operacional.

3) Suponemos que en el lenguaje Pam añadimos una instrucción de asignación múltiple:

I1, I2 := E1, E2 (donde $I1 \neq I2$)

Describe formalmente su semántica, tanto en el caso en que las asignaciones se realicen de forma secuencial como en el caso en que se produzcan en paralelo. Da además un ejemplo que muestre la diferencia.

4) Dados los dos programas en Eva:

a) begin char c input c output c end	b) begin string s proc p = begin char c input c cons c, s end call p output head s end
---	---

¿Son semánticamente equivalentes desde el punto de vista denotacional?

¿Y operacionalmente?

5) Suponemos que en el lenguaje Eva además de las declaraciones usuales, se permite declarar identificadores con un valor inicial, por ejemplo:

```

char I : "f"
string S : "casa"

```

Modifica la semántica denotacional de Eva para que incluya estas nuevas declaraciones.

6) Modifica la semántica denotacional del lenguaje Eva para incluir el uso de funciones. Utiliza la siguiente modificación de la sintaxis abstracta:

$$\Delta ::= \dots \mid \text{fun } I = E \mid \text{fun } I (T1 \ I1, \dots, Tn \ In) = E \mid \dots \quad (\Delta = \text{declaraciones})$$

$$E ::= \dots \mid \text{callf } I \mid \text{callf } I (E1, \dots, En) \mid \dots \quad (E = \text{expresiones})$$

Nota: Se deben definir los dominios semánticos y las ecuaciones semánticas necesarias.

7) Modifica la semántica denotacional de Eva para que el paso de parámetros sea por "**por referencia**" en lugar de "**por valor**". Explica el cambio realizado.

8) Modifica la semántica denotacional de Eva para que el paso de parámetros sea por "**valor-resultado**" en lugar de "**por valor**". Explica el cambio realizado.

9) ¿Qué habría que modificar en la semántica denotacional de EVA si los identificadores no locales dentro de un procedimiento se ligan *dinámicamente*, es decir, en el entorno de la llamada al procedimiento?

10) Supongamos un lenguaje de programación donde la habitual sentencia "read" sea considerada una expresión, de modo que su evaluación produce un valor y la modificación del fichero de entrada como efecto lateral. Partiendo de la siguiente sintaxis abstracta para las expresiones:

$$\begin{aligned} I &: \text{Ide} \\ E &: \text{Exp} \\ E &::= 0 \mid 1 \mid \text{read} \mid I \mid E1 + E2 \end{aligned}$$

Define el significado de una expresión E por el método denotacional.

11) El significado de las declaraciones en Eva (tabla 4.4) se ha definido mediante la función semántica

$$D: \text{Dec} \rightarrow U \rightarrow (U * S) \rightarrow (U * S)$$

en lugar de definirse mediante la función más simple $D: \text{Dec} \rightarrow (U * S) \rightarrow (U * S)$.

Razona el motivo y acompaña tu explicación con un ejemplo ilustrativo.

12) La instrucción repeat con formato **repeat I until C end** (siendo I una instrucción y C una condición booleana) es semánticamente equivalente a la siguiente secuencia de instrucciones:

$$\begin{aligned} &I; \\ &\text{while (not C) do } I \text{ end} \end{aligned}$$

Define, desde el punto de vista denotacional, la semántica de la instrucción **repeat** en términos recursivos, razonándola (sintetizándola) a partir de la equivalencia dada.