

Autómatas y Lenguajes. Ejercicios sobre Gramáticas de Atributos.

Considérese la siguiente gramática independiente del contexto:

```
axioma      ::= USUARIO peliculas
peliculas   ::= peliculas pelicula
              |  pelicula
pelicula     ::= TITULO valoraciones
valoraciones ::= valoraciones valoracion
              |  valoracion
valoracion   ::= (USUARIO:NUMERO)
```

Esta gramática genera frases que consisten en una secuencia de películas con sus respectivas valoraciones realizadas por los usuarios. Las valoraciones son números naturales entre 1 y 100. Al principio de las frases generadas por la gramática siempre aparece el nombre de un usuario. A continuación se muestran tres ejemplos:

```
jaime Star_Wars(jaime:10)(ana:10) Spiderman(ana:10)(pepe:1)
```

```
ana Kapax(pepe:100)(jaime:1) Avatar(ana:40) Pi(pepe:100)(ana:1)
```

```
mike Kapax(pepe:100)(jaime:1) Avatar(ana:40) Pi(pepe:100)(ana:1)
```

Se desea añadir a esta gramática un sistema de atributos que haga lo siguiente:

1. Calcular el número total de películas valoradas.
2. Indicar el título y la valoración de la película mejor valorada por el usuario indicado al principio de la cadena. Si no hay ninguna película valorada por dicho usuario debe indicarse tal circunstancia.

Por ejemplo, para las tres cadenas anteriores se mostrarían respectivamente los mensajes siguientes:

```
Se han valorado 2 peliculas.
```

```
La mejor valorada por jaime es Star_Wars, con una valoración de 10.
```

```
Se han valorado 3 peliculas.
```

```
La mejor valorada por ana es Avatar, con una valoración de 40.
```

```
Se han valorado 3 peliculas.
```

```
Ninguna pelicula ha sido valorada por mike.
```

NOTAS:

- Puede considerarse que existe un proceso de análisis morfológico que asigna a los símbolos terminales los siguientes atributos semánticos: `USUARIO.nombre` (nombre del usuario), `TITULO.titulo` (título de la película), `NUMERO.valor` (valor numérico del número).
- Debe comprobarse que el usuario que aparece al principio de la cadena no valora más de una vez la misma película. No es necesario realizar esta comprobación para el resto de usuarios.
- No está permitido el uso de información global.
- Debe responderse a cada una de las cuestiones planteadas usando las plantillas correspondientes.

Ejercicio 1. Describe explícita y brevemente el significado de cada atributo que utilices, así como el proceso de actualización de cada uno de ellos mencionando si se realiza herencia o síntesis en su propagación. Utiliza la siguiente tabla.

SÍMBOLO	ATRIBUTOS	DESCRIPCIÓN
USUARIO	nombre	Nombre del usuario. Calculado por el analizador morfológico.
TITULO	titulo	Título de la película. Calculado por el analizador morfológico.
NUMERO	valor	Valor numérico del número. Calculado por el analizador morfológico.
valoracion		
valoraciones		
pelicula		
peliculas		

Ejercicio 2. Define formalmente la gramática utilizando la notación explicada en el temario de esta asignatura. Utiliza la siguiente tabla, en la que debes indicar, para cada regla de la gramática, las acciones semánticas asociadas junto con el instante en que se deben ejecutar.

INSTANTE	ACCIÓN
axioma ::= USUARIO peliculas	
peliculas ₁ ::= peliculas ₂ pelicula	
peliculas ::= pelicula	
pelicula ::= TITULO valoraciones	

valoraciones ₁ ::= valoraciones ₂ valoracion	
valoraciones ::= valoracion	
valoracion ::= (USUARIO:NUMERO)	

Ejercicio 3. La tabla que aparece a continuación representa el árbol de derivación para la siguiente sentencia:

```
jaime Star_Wars(jaime:10)(ana:30) Spiderman(ana:10)(jaime:20)
```

Nótese que, para simplificar, se están ignorando los terminales (, : y) en la regla para el no terminal valoracion. Muestra, sobre la tabla, los valores de los atributos semánticos y alguna indicación gráfica (por ejemplo flechas) que indique cómo se realiza la propagación.

axioma												
USUARIO	peliculas											
	peliculas					pelicula						
	pelicula											
	TITULO	valoraciones					TITULO	valoraciones				
		valoraciones			valoracion			valoraciones			valoracion	
		valoracion						valoracion				
		USUARIO	NUMERO	USUARIO	NUMERO	USUARIO		NUMERO	USUARIO	NUMERO	USUARIO	NUMERO
jaime	Star_Wars	jaime	10	ana	30	Spiderman	ana	10	jaime	20		

Ejercicio 4. Considérese la siguiente gramática independiente del contexto:

```
axioma      ::= DINERO lista_compra
lista_compra ::= compra lista_compra
              | compra
compra       ::= [PRODUCTO, PRECIO]
```

Esta gramática genera sentencias que consisten en una cantidad de DINERO seguida de una lista de compras, cada una con un PRODUCTO y el PRECIO del mismo. Tanto el DINERO como el PRECIO son números reales. A continuación se muestran algunos ejemplos:

```
50.00 [pan, 0.90] [leche, 1.50] [tomates, 1.20]
17.00 [huevos, 1.5] [cafe, 1.60] [arroz, 1.80] [vino, 8.50] [chocolate, 2.00]
10.00 [cervezas, 4.20] [pizza, 3.40] [aceitunas, 1.30] [patatas, 1.50]
```

Se desea añadir a esta gramática un sistema de atributos que compruebe si se puede hacer la compra con el dinero disponible. Se indicarán los productos de la lista que se pueden comprar y el saldo disponible al final. Se supone que los productos se adquieren en el mismo orden en el que aparecen en la lista. Por ejemplo, para las tres sentencias anteriores se mostrarían respectivamente los mensajes siguientes:

```
disponible: 50.00
  pan: 0.90
  leche: 1.50
  tomates: 1.20
  total: 3.60
  saldo: 46.40
```

```
disponible: 17.00
  huevos: 1.50
  cafe: 1.60
  arroz: 1.80
  vino: 8.50
  chocolate: 2.00
  total: 15.60
  saldo: 1.60
```

```
disponible: 10.00
  cervezas: 4.20
  pizza: 3.40
  aceitunas: 1.30
  total: 8.90
  saldo: 1.10
```

NOTAS:

- Puede considerarse que existe un proceso de análisis morfológico que asigna a los símbolos terminales los siguientes atributos semánticos: PRODUCTO.nombre (nombre del producto), PRECIO.valor (valor numérico del precio) y DINERO.valor (valor numérico del dinero).
- No está permitido el uso de información global.

Ejercicio 5. Considérese la siguiente gramática independiente del contexto:

```
romano ::= IList
        | I V
        | V IList
IList  ::= IList I
        | λ
```

Esta gramática genera sentencias entre las que se encuentran los números romanos hasta el 8 (deben ignorarse los espacios). A continuación se muestran dos sentencias generadas por la gramática:

```
VII
IIII
```

Se pide añadir a la gramática un sistema de atributos que compruebe si la sentencia corresponde a un número romano correcto, y en caso afirmativo imprima su valor numérico. Por ejemplo, para las dos sentencias anteriores el resultado sería, respectivamente:

```
VII, número romano correcto, su valor es 7
IIII, no es un número romano correcto
```

No está permitido el uso de información global.

Ejercicio 6. Considérese la gramática independiente del contexto resultante de añadir a la del ejercicio 5 el siguiente axioma:

```
S ::= romano = IList
```

Esta gramática genera sentencias formadas por dos cadenas separadas por el signo igual (=). La primera cadena (a la izquierda del igual) es igual que las generadas por la gramática del ejercicio 5. La segunda cadena (a la derecha del igual) es una lista de símbolos I. Algunas sentencias generadas por esta gramática son las siguientes:

```
VII = IIIIIII
VII = III
IIII = IIII
```

Se pide añadir a la gramática un sistema de atributos que compruebe si, en el caso de que la primera cadena sea un número romano correcto, el valor del mismo coincide con el número de símbolos I en la segunda cadena. Por ejemplo, para las sentencias anteriores el resultado podría ser:

```
VII, número romano correcto, su valor es 7, coincide con segunda cadena
VII, número romano correcto, su valor es 7, no coincide con segunda cadena
IIII, no es un número romano correcto
```

No está permitido el uso de información global.