



## Parcial 02 Recuperatorio 02

1) Encuentre el conjunto  $\text{Siguiente}(M)$  para la siguiente gramática

$$S \rightarrow AMbT$$
$$A \rightarrow pT$$
$$M \rightarrow rP$$
$$T \rightarrow xMA$$

### Solución

$$\text{Siguiente}(M) = \{b, p\}$$

2) Dada la siguiente GIC encuentre una equivalente LL(1)

$$S \rightarrow \text{reCup} \mid \text{resT} \mid \text{rey}$$
$$T \rightarrow \text{Toro} \mid \text{mu}$$

### Solución

$$S \rightarrow \text{reS}'$$
$$S' \rightarrow \text{Cup} \mid sT \mid y$$
$$T \rightarrow \text{muT}'$$
$$T' \rightarrow \text{oroT}' \mid \varepsilon$$

3) Escriba el PAS correspondiente a la siguiente producción.

$$\langle \text{Elegir} \rangle ::= \text{UNO} \langle \text{Datos} \rangle \text{DOS} \mid \text{OTRO} \langle \text{Vuelta} \rangle \mid \text{ALTER}$$

### Solución

```
void Sentencia()
{
    switch(ProximoToken()) {
        case UNO:
            Match(UNO);
            Datos();
            Match(DOS);
            break;
        case OTRO:
            Match(OTRO);
            Vuelta();
            break;
        case ALTER:
            Match(ALTER);
            break;
        default:
            ErrorSintactico();
    }
}
```

4) Dado el siguiente fragmento del estándar de C

additive-expression:

    multiplicative-expression

    additive-expression + multiplicative-expression

    additive-expression - multiplicative-expression

shift-expression:

    additive-expression

    shift-expression << additive-expression

    shift-expression >> additive-expression



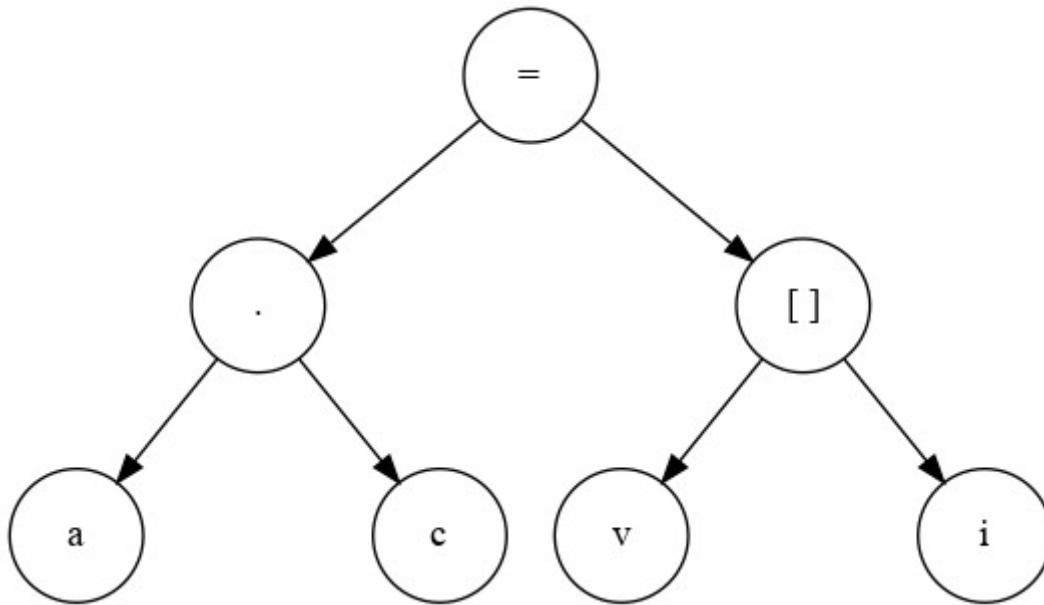
a) Indique, justificando, como asocia el operador  $>>$ .

De izquierda a derecha, porque la recursión es a izquierda

b) Indique, justificando, que operador tiene mayor precedencia entre  $<<$  y  $+$ .

El operador  $+$  tiene mayor precedencia que  $<<$  porque está más lejos del axioma.

5) Dibuje el árbol de la expresión en lenguaje C:  $a.c = v[i]$



6) Dada la siguiente expresión en lenguaje C:

$p \rightarrow e.v[i]$

a) Haga las declaraciones necesarias para que sea semánticamente correcta y de tipo unsigned char.

```
int i;
struct A {
    struct B { unsigned char v[5]; } e;
} *p;
```

b) Enuncie una condición para que sea pragmáticamente correcta.

Que p apunte a una zona de memoria correctamente reservada y que i tenga un valor dentro del rango del arreglo v.

7) Indique en una frase el tipo de datos de r, de acuerdo a la siguiente declaración

$\text{int}^* (*r)[5];$

r es un puntero a un arreglo de 5 punteros a enteros.

8) Dado el siguiente fuente

```
47 int calculo(int param) {
48     static int nivel;
49     while(param>0) {
50         param -= param/2 + nivel;
51         nivel++;
52     }
```



```
53     return param * -nivel;  
54 }
```

Conteste las siguientes preguntas.

**a)** A qué categoría sintáctica pertenece lo escrito entre los paréntesis en la línea 49?

Expresión.

**b)** De la lista de caracteres que serán devueltos con `ungetc` por el scanner en la línea 49.

( , > , 0 , )

**c)** Cuanto devuelve la función, la primera vez que se invoque en el programa, si le pasan -7 como valor de `param`. Justifique

Cero, porque nunca entra al `while` y `nivel` al ser estático la primera vez vale cero.