

PROCESADORES DEL LENGUAJE

Primera Aproximación:



Grupo: 113-leal-zhu

ID de grupo de prácticas: 81

Nombre de los participantes:

- Liang Ji Zhu
- Ignacio Leal Sánchez

Correos electrónicos de los participantes:

- 100495723@alumnos.uc3m.es
- 100495680@alumnos.uc3m.es

Primera Aproximación

En esta primera aproximación tenemos que hacer un traductor que transformase la notación infija a la notación postfija. Hemos logrado el dicho resultado sustituyendo la lógica por la impresión de los caracteres que aparecen por cómo se ejecuta el código proporcionado, ya que la operación se ejecutará una vez se haya ejecutado todas sus hojas, y por ello se imprimirá en postfija directamente. Solo se necesitó cambiar el orden de la impresión, ya que se trataba de un árbol ascendente.

```
C/C++
// SECCION 3: Gramatica - Semantico
axioma:      expresion '\n'          { printf (".\n") ; }
           r_expr
           | VARIABLE '=' expresion '\n' {
                                   printf ("%c !\n", 'A' + $1);
                                   }
           r_expr
           ;

r_expr:      /* lambda */
           | axiom
           ;

expresion:    termino                { $$ = $1 ; }
           | expresion '+' expresion  { printf( "+ "); }
           | expresion '-' expresion  { printf( "- "); }
           | expresion '*' expresion  { printf( "* "); }
           | expresion '/' expresion  { printf( "/" ); }
           ;

termino:      operando                { $$ = $1 ; }
           | '+' operando %prec SIGNO_UNARIO { printf( "%d ", $2); }
           | '-' operando %prec SIGNO_UNARIO { printf( " negate"); }
           ;

operando:     VARIABLE                { printf("%c@ ", 'A' + $1) ; }
           | NUMERO                   { printf( "%d ", $1); }
           | '(' expresion ')'         { printf( "(%d ) ", $2); }
           ;
```

Hemos comprobado que los resultados traducidos son interpretables en el lenguaje Forth.

```
C/C++
B=A*10+23/10
A@ 10 * 23 10 / + B !
-----
6 * B + 1 * 2
6 B@ * 1 2 * + .
```

- 3

3 negate .

1 +1

1 1 + .

A = 6

6 A !
