

Analizador Sintáctico

Backus–Naur Form:

- $S::= A\{[K]A\}$
- $K::= + \mid * [+]$
- $A::= a \mid \# \mid \backslash \mid (S)$

Nota: a = Alfabeto en minúsculas; $\#$ = dígitos $\{0, 1, \dots, 9\}$; \backslash = lambda

Nota: El no-terminal K no es necesario en el algoritmo, solo se colocó para facilitar su entendimiento

Algoritmo de Parsing:

```
procedure Parser()  
    nextToken();  
    PS();  
    expect($);  
end
```

```
procedure PS()  
    PA();  
    While Token in  $[+|-]$  do  
        Case Token in  
             $[+]$ : sigToken(); PA();  
             $[*]$ : sigToken(); If (Token = +) then sigToken(); fi PA();  
            Otherwise: error();  
        Esac  
    done  
end
```

```
procedure PA()  
    Case Token in  
         $[a]$ : sigToken();  
         $[\#]$ : sigToken();  
         $[\backslash]$ : sigToken();  
         $[()]$ : sigToken(); PS(); expect( ) );  
        Otherwise: error();  
    Esac  
end
```