

## Tarea Programada Número 2.

En esta tarea usted debe hacer un intérprete de Mini-Pascal en C o C++, para ello debe:

1. Implementar un analizador léxico y tokenizador para este lenguaje que genere un stream de tokens.
2. Implementar un analizador sintáctico para el lenguaje que tome como entrada el stream de tokens y genere un árbol sintáctico.
3. Implementar un analizador contextual que tome como entrada el árbol sintáctico del punto 2 y genere un árbol sintáctico anotado.
4. Implementar intérprete que ejecute el árbol sintáctico.

La interfaz de su programa debe ser por línea de comandos:

```
>> pascal programa.pas
```

```
Hola Mundo!
```

Donde su programa es:

```
program programa;  
begin  
    write("Hola Mundo!");  
end.
```

Otro ejemplo de mini-pascal es:

```
program foo(boo);  
    // variable globales  
    var a, b, c, d: integer;  
  
    // declaración de funciones  
    function addition(a, b: integer) : integer;  
        var c: integer;  
        Begin  
            return a + b  
        end;  
  
    // compound statements: "main"  
    Begin  
        a := addition(1, 2);  
        putchar(a)  
    end.
```

Información para implementar mini-pascal la puede encontrar en:

<https://github.com/banacorn/mini-pascal>

La sintaxis de mini-pascal y la semántica de pascal completo están en dos documentos adjuntos.

### Notas

- El mini pascal tiene una sintaxis sencilla que pueden implementar utilizando un parser por descenso recursivo, SLR, o cualquiera de las técnicas que hemos visto en clase.
- Su programa debe dar información sobre errores de una manera razonable (dar feedback al programador que indique donde se encuentra el error y cuál es la posible razón.
- No hace falta que su programa acumule errores, con que pare en el primer error que se encuentre y lo reporte es suficiente.
- Separe completamente las rutinas que hagan las distintas fases de compilación, esto es: análisis léxico, análisis sintáctico, análisis contextual, e interpretación. Estas fases se evaluarán independientemente.
- Para la siguiente tarea usted debe sustituir la fase de interpretación por un generador de código. Trate de que su tarea sea clara para que el cambio de componente no le resulte difícil de hacer.
- Para la siguiente tarea usted debe agregar procedimientos y funciones al mini-pascal, así que de nuevo, trate de que su código sea modular para que pueda agregar este patrón al analizador sintáctico/contextual.

Fecha de entrega: 17 de Mayo.