

Universidad Simón Bolívar

Departamento de Computación y Tecnología de la Información

CI3725 - Traductores e Interpretadores

Diciembre - Marzo 2014-2015

Setlan - Etapa IV

Interpretador con verificaciones dinámicas (12%)

Especificación de la entrega

Finalizada la construcción del árbol abstracto sintáctico y la tabla de símbolos, corresponde la implementación del interpretador *final* de programas en **Setlan**. Su programa principal deberá recorrer el árbol sintáctico abstracto, ejecutar las instrucciones y evaluar las expresiones encontradas.

Para la clase correspondiente a las instrucciones del programa deberá implementar un procedimiento **execute** (o **ejecutar** si está programando en castellano) y para la clase que describe expresiones del lenguaje deberá implementar un procedimiento **evaluate** (o **evaluar**). Para esta entrega no debe imprimirse información acerca del interpretador (tabla de símbolos, lista de tokens, etc.); sino debe correrse e imprimir lo que indique el programa en **Setlan**, es decir, imprimir por salida estándar sólo aquellas expresiones que son argumento de las instrucciones **print** y **println**.

En tiempo de ejecución se deben considerar estas verificaciones dinámicas para garantizar una ejecución *limpia* del programa interpretado:

- Errores de división por cero

- Errores de división por cero.
- Errores de *overflow*.
- Errores de máximo y mínimo elemento de un conjunto vacío.

Si existen errores resultado del análisis estático, deberán ser impresos por salida estándar siguiendo el formato especificado en las entregas anteriores y abortar la ejecución del intérprete.

Ejecución

Para la ejecución del interpretador se programa deberá llamarse `setlan` y recibirá como primer argumento el nombre del archivo con el código en `Setlan` a analizar. La salida del programa dependerá *únicamente* de las instrucciones `read`, `print` y `println`.

Ejemplo de impresión de errores:

Programa:

```
program
  for i min {4,2,1,2,5} do
    print i % (5 - i), " "
```

Salida correspondiente:

```
1 2 0
```

```
ERROR: division by zero in operation at line 3, column 9
```

Programa:

```
program {
  using
    int max_int, n;
  in
```

```
    LFI

    max_int = 2147483646;
    n = 2*max_int;
}
```

Salida correspondiente:

```
ERROR: overflow in operation at line 7, column 9
```

Programa:

```
program {
    using
        int max_e;
    in

    max_e = >? {};
    println max_e;
}
```

Salida correspondiente:

```
ERROR: empty set in operation at line 6, column 13
```

Implementación

Para la implementación del interpretador del lenguaje **Setlan**, pueden escoger uno (1) de los tres (3) lenguajes de programación a continuación. Para cada uno de ellos se indica las herramientas disponibles para el desarrollo de un interpretador de código. Recuerden que para esta etapa de desarrollo deberán seguir empleando **el lenguaje de programación utilizado en las etapas anteriores a ésta**.

programacion utilizado en las etapas anteriores a esta.

- *Python:*
 - Interpretador *python* 2.7.
 - *Python Lex-Yacc (PLY)*.
- *Ruby:*
 - Interpretador *ruby* 1.9.
 - *Racc*.
- *Haskell:*
 - Compilador *ghc* 7.6.3 ó 7.8.3.
 - *Alex* y *Happy*.

Formato de Entrega:

Deben enviar un correo electrónico a **todos los preparadores** con el asunto:

[CI3725]eXgY donde X corresponde al número de la entrega e Y al número del equipo. El correo debe incluir un archivo comprimido **.zip** que contenga:

- Código fuente debidamente documentado.
- En caso de utilizar Haskell, deben incluir un archivo Makefile o cabal. Si su proyecto no compila, el proyecto no será corregido.
- Un archivo de texto con el nombre **LEEME.txt** donde **brevemente** se expliquen:
 - Estado actual del proyecto.
 - Problemas presentes.
 - Cualquier comentario respecto al proyecto que consideren necesario.
 - Este archivo debe estar identificado con los nombres, apellidos y carné de cada miembro del equipo de trabajo.

Recuerde que no cumplir con estas especificaciones puede afectar directamente la nota final de la entrega.

Fecha de entrega:

La fecha límite de entrega del proyecto es el día **viernes 06** de marzo de 2015 (semana 11) hasta las **11:50pm**. Entregas hechas más tarde tendrán una **penalización del 20%**.

11) hasta las 11:00pm. Entregas hechas más tarde tendrán una **penalización del 20 %** de la nota. Esta penalización aplica por cada día de retraso.