

Tarea Programada III

Elizabeth Córdoba

Eduardo Álvarez

Andrés Abarca

Josué Fallas

Marianela Cordero

07 de noviembre del 2013

Tabla de Contenidos

Descripción del problema.....	02
Diseño del programa.....	03
Librerías usadas.....	05
Análisis de resultados.....	06
Conclusion personal.....	07
Bibliografía.....	08

Descripción del problema

La tercera tarea programada tiene como objetivo la práctica de algunos conceptos teóricos de lenguajes de programación, mediante la implementación de un manejador de tablas de símbolos para lenguajes de programación, en nuestro caso en lenguaje de programación python y el sistema operativo Ubuntu.

El proyecto se divide en 2 secciones, el ambiente estático y el ambiente dinámico:

- Para la tabla del ambiente estático, deberán almacenar la información de los tipos de datos de cada una de las expresiones, esta se imprimirá en consola determinando el tipo de valor de una variable determinada, en nuestro caso esta evaluará integers, booleans, listas y tuplas.
- Para la tabla del ambiente dinámico, deberán almacenar la información de los valores de cada una de las expresiones, expresando el resultado final o lo que contiene específicamente la variable, también se evaluará integers, booleans, listas y tuplas.

Para llevar a cabo cada una de las partes mencionadas anteriormente se debe investigar el correcto uso y evaluación de programas por medio del ambiente estático y dinámico propiamente en Linux.

Cabe destacar que el programa no usa interfaz grafica sino que trabaja propiamente en consola.

Diseño del programa

Ambiente Estático:

Para la evaluación del ambiente estático decidimos almacenar cada expresión leída del archivo en una lista.

La evaluación está diseñada por etapas, es decir, primeramente evalúa expresiones sencillas (cuando la expresión no se compone de variables), luego se evalúan expresiones formadas por variables, las cuales se deben buscar ya que ya han sido evaluadas anteriormente. Luego se evalúan tuplas y listas, así como sus posibles combinaciones. Y por último se evalúan los lets y los ifs.

Para estos últimos, se tiene el cuidado de respetar el alcance de las expresiones dentro de ellos, ya que fuera de estos esas variables no existen, al menos que el resultado de estas expresiones se almacene en una variable de manera explícita.

En la tabla se imprimen todos los tipos y las variables, aunque estos se encuentren dentro de los lets y los ifs, pero siempre se respetan sus alcances, es decir el tipo de una expresión let o if no afecta expresiones futuras si este no existe fuera de ellas.

Ambiente Dinámico:

Para la implementación de la tabla dinámica se decidió utilizar una lista para poder almacenar el código segmentado por la condición "Val x =", de esta forma la función Procesar_Exp() podrá diferenciar si la variable es un valor constante, un let, un if, un Boolean, esta diferenciación se realiza mediante ifs, en caso de que la variable contenga un if o un let se hará el

llamado respectivo a `Procesar_if()` o `procesar_let()`. Además en caso de que la variable lo que contenga sea una operación también se procederá a hacer el llamado a la función `operación ()` que se encargará de preparar operadores y operando y así `operar2num ()` podrá calcular el resultado final. El retorno de cada `if` de `Procesar_Exp()` será almacenado en una lista llamada `tabla` para posteriormente ser impresa con el encabezado de `tabla` dinámica.

Librerías Utilizadas

- **Math:** esta librería se utilizo, para tener cierta seguridad en algunas actividades matemáticas que utilizaremos en el ambiente dinámico, si bien es cierto esto se realiza como un respaldo sobre acciones o operaciones matemáticas, importando todas las operaciones correspondientes a esta librería, que son una infinidad, pero en si se utilizarán específicamente suma, resta, multiplicación, división, mod y div propiamente del lenguaje Python.

Análisis de resultados

A continuación se especificaran los objetivos que se cumplieron en el proyecto programado III:

- Se logró leer cualquier archivo de texto, siempre y cuando dentro de él se encuentre código del lenguaje SML.
- Se logró implementar la tabla de ambiente estático, con sus validaciones respectivas, además este ambiente tiene un grado de robustez alto, ya que se probó con gran cantidad de ejemplos y maneras, logrando la validación completa de acuerdo a lo solicitado en el enunciado de la tarea programada 3.
- Se logró implementar la tabla de ambiente dinámico, con sus respectivas validaciones, imprimiendo su resultado en conjunto con el ambiente estático de forma ordenada en consola.

Objetivos que no se cumplieron:

- El ambiente dinámico no es del todo robusto ya que no se logró procesar if y let, además de operaciones con variables definidas, e intercambiar variables por sus respectivos valores, todo esto por falta de tiempo al realizar el trabajo.

Conclusión personal

En la Tarea Programada III, ponemos a prueba todos nuestros conocimientos sobre programación, que hemos adquirido en otros cursos, además enfrentarnos al reto de implementar un lenguaje de programación funcional como lo es SML, en otro orientado a objetos, en este caso Python. También trabajamos con el sistema operativo Linux, específicamente con Ubuntu, el cual ha sido nuestra tercera experiencia desarrollando en este sistema operativo.

Durante la creación o desarrollo de esta tarea implementamos algo distinto, que no solo se sale de algo estándar, sino que es un implemento e idea brillante, implementar un lenguaje escrito básicamente en otro lenguaje de programación, con su misma sintaxis e implementación lógica.

Esto nos permitió determinar conocimientos específicamente de SML, logrando el análisis y evaluación de un ambiente estático y dinámico de este lenguaje interpretándolo y imprimiéndolo en otro lenguaje, además del conocimiento y practica que nos brinda la implementación, sintaxis, análisis (léxico, sintáctico), que se tuvieron que entender y desarrollar.

En el desarrollo de esta tarea, reforzamos nuestra habilidad de trabajo en equipo ya que al trabajar en un grupo de 5 personas se dificulta un poco poder organizar las reuniones y poder adaptar los distintos puntos de vista de manera que se combinen de la mejor manera para obtener el resultado esperado. Pero al poder concluir exitosamente esta tarea, logramos superar la prueba y adquirimos experiencia tanto en programación como en habilidades interpersonales, para saber trabajar en equipo.

Bibliografía

Python, D. (s.f.). *Mathematical functions*. Obtenido de <http://docs.python.org/2/library/math.html>