

Tarea Programada 3

Documentación

Externa

INSTITUTO TECNOLÓGICO DE COSTA RICA

INTEGRANTES:

EDGAR SOLÓRZANO PACHECO

DANIEL MORA CHACÓN

GERARDO CALDERÓN BADILLA

PROFESOR: ANDREI FUENTES LEIVA

Resumen ejecutivo

Esta tarea programada tiene como objetivo recrear la codificación de los datos del ambiente estático y dinámico del Lenguaje de Programación SML, desarrollando un programa en un lenguaje alternativo y de libre escogencia. Los resultados de esta tarea serán generados en una tabla que se mostrarán mediante una aplicación web.

Como ambiente estático se entiende aquel que almacena los tipos de variables argumentos de las expresiones que maneja el lenguaje de programación SML. Con respecto al ambiente dinámico, este es aquel que maneja los valores de las expresiones que contiene el archivo.

La tarea programada fue desarrollada en Java NetBeans o Eclipse, ya que este lenguaje fue el más fácil de utilizar debido a la experiencia adquirida anteriormente. Para el funcionamiento de la tarea programada se definieron en Java diversas funciones que verifican y validan el manejo de las expresiones del lenguaje SML

Para la muestra de los resultados, este se realiza mediante una tabla que muestra la expresión, los tipos de esas expresiones (ambiente estático) y los valores de esas expresiones (ambiente dinámico). Todo esto será manejado por una aplicación web que permita cargar el archivo SML y retorne los ambientes mencionados anteriormente.

Propósito

Descripción del Aplicación

El propósito de la aplicación, es realizar una simulación de los ambientes estático y dinámico del lenguaje SML. Para mostrarlo al usuario mediante una tabla que contenga los valores y los tipos de expresiones, se utilizará una aplicación web.

En la aplicación web tiene incorporado unos botones que permitirán cargar el archivo, y generar la información de los ambientes en una tabla.

Requerimientos

Para esta tarea el lenguaje es opcional, es decir, a escogencia de los programadores. Para este caso se utilizará Java, bajo el entorno de NetBeans por lo que es recomendable utilizar este programa para correr la aplicación.

En cuanto al sistema operativo donde se correrá el programa, lo más recomendable es usar Ubuntu, pero si desea puede utilizar las versiones más recientes de Windows (Windows 7, Windows 8.1) para ejecutar el programa.

Funcionalidades

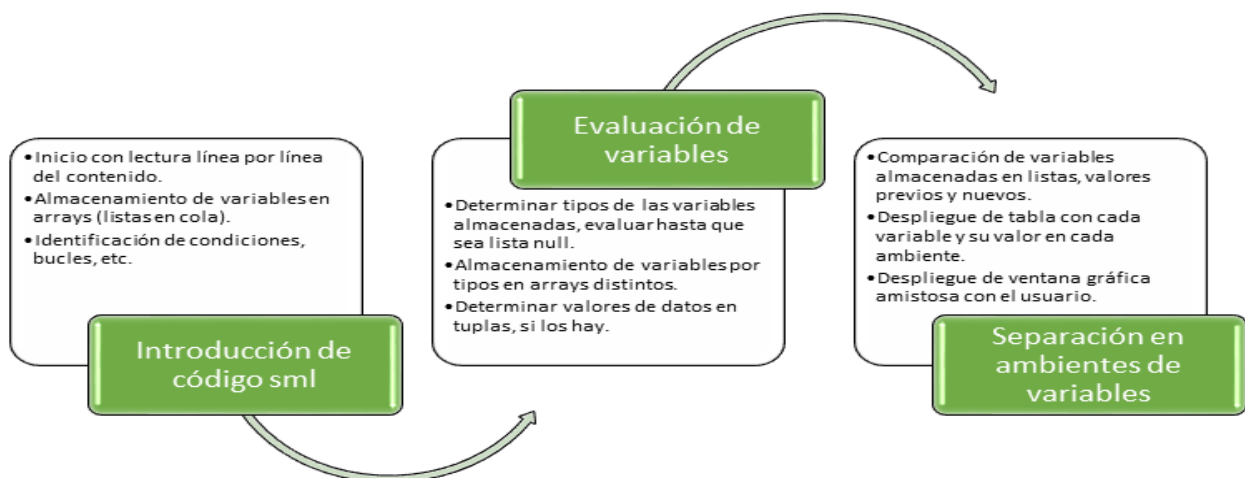
1. **Cargar el archivo SML:** Mediante la aplicación web, esta tendrá una función que le permita al usuario cargar el archivo SML para la toma de datos, para posteriormente retornar la información con el tipo de expresiones y los valores correspondientes.
2. **Lectura de código del archivo SML:** El programa es capaz de leer línea por línea, el archivo para examinar cada expresión y así poder generar la tabla que muestre el tipo y los valores de las expresiones.
3. **Evaluar y determinar las expresiones:** Una vez que el programa lee el archivo, se va analizando las líneas del archivo, identifica tanto los tipos, como los valores, según las expresiones. Una vez hecho esto, el programa ejecuta una serie de funciones para obtener los resultados de dichas expresiones.
4. **Desplegar los resultados en la aplicación web:** Una vez analizada cada expresión y determinado los diferentes tipos y valores, el programa despliega una tabla con los ambientes estático y dinámico en una aplicación web.

Descripción de diseño de alto nivel

- **Decisiones de Diseño**

Se decidió desarrollar el programa en el lenguaje de programación Java, porque el equipo de trabajo tiene más experiencia en la creación de código, además su facilidad permite agilizar el desarrollo del programa.

Es un lenguaje fácil de entender y leer, asimismo cuenta con más documentación en la web, ya que su uso se extiende en la mayoría de programadores. Además permite la flexibilidad debido a que es posible encontrar gran cantidad de librerías y herramientas para conseguir el objetivo deseado. Se utilizó el despliegue de datos por medio de una tabla, ya que la visualización y el entendimiento del código, depende de cómo esté el código en SML, entonces, la información se despliega ordenadamente, de acuerdo al código.



- **Funciones utilizadas**

LecturaArchivo: Función que lee los archivos SML. Recibe un nombre de un archivo para poder leer el archivo a utilizar

Separar: Ayuda a separar los espacios en cada línea y guarda en un arreglo cada palabra de las expresiones del archivo. Recibe una expresión del archivo y separa cada línea

Validar: Encargada de analizar las expresiones para obtener los valores y los tipos

Determinar_Valor: Esta función es la encargada de asignar el tipo de dato de dato a la que pertenece cada palabra, es decir verifica el tipo de dato del ambiente estático. Recibe el arreglo donde se almacenó las expresiones

ResultadoExpresion: Función encargada de realizar las operaciones cuando encuentre operandos y valores en las expresiones

Comparar: Función que compara dos valores y retorna “true” o “false”, según sea el caso.

Valores_Tuplas / Valores_Listas: Funciones encargadas de determinar los tipos de datos que tienen internamente estos arreglos, para posteriormente mostrarlos en la tabla.

- **Librerías usadas**

- **java.io:** Librería que permite acceder al listado de las unidades del sistema de ficheros
- **java.lang:** Se importa automáticamente y posee gran parte de las clases más utilizadas.
- **java.util.Map :** Utilizada para el mapeo de los parámetros por el URL.
- **java.servlet:** Librería encargada de permitir la configuración y diseño del Servlet controla la aplicación.

- ***Lenguajes de Programación Usados***

Java. Debido a la experiencia adquirida en el curso Algoritmos y Estructuras de Datos el desarrollo de la tarea programada será implementado en lenguaje Java para mayor simplicidad y más facilidad al aplicar los conocimientos en este lenguaje.

Problemas encontrados

- Creación de listas anidadas, listas con tuplas o expresiones para evaluar “let” con más de una variable a utilizar dentro del “in”, da un error que no pudimos corregir, por lo tanto para su solución se propone una mejor implementación sin tantos ciclos para los diferentes tipos de datos como por cada una de las líneas.
- Lectura de archivos se realiza con buffereader

Manual de usuario

1. Instalar la última versión de Netbeans.
2. Importar el proyecto (aplicación web Ambiente)
3. Ejecutar el proyecto
4. Hacer clic en el botón Cargar archivo para especificar el archivo SML que se va a cargar
5. Hacer clic en el botón Analizar
6. A continuación se muestra el resultado del análisis.
7. Fin del programa.