|  |  |
| --- | --- |
| ***Instituto Tecnológico de Costa Rica***  ***Sede Regional San Carlos*** | Tercer Tarea Programada  Compiladores e Intérpretes |
| Prof. Oscar Mario Víquez Acuña | ***Generador de Código*** |

**Descripción:**

Esta etapa es el punto máximo en el desarrollo del compilador. Es en esta fase donde a partir de los resultados correctos de las fases anteriores se comienza con la tarea de generar las instrucciones de máquina equivalentes a la estructura de programa almacenada en el AST.

Para realizar esta etapa será necesario generar código para la máquina virtual simple que se detalla en el documento adjunto. Esta máquina virtual contempla solamente algunos aspectos funcionales del lenguaje, por lo que será necesario9 ampliar sus capacidades para lograr la funcionalidad de todos los aspectos denotados en este documento.

La técnica para la generación de código será el recorrido mediante “visitors” de manera que el código se irá generando según el orden establecido por las visitas en un recorrido en profundidad del árbol.

Se deben implementar o utilizar en el lenguaje de implementación, las funciones que permitan la ejecución de los métodos predefinidos documentados en las etapas anteriores (métodos para listas, imprimir, etc).

**Tipos de datos y estructura del código:**

Tipos de datos

* Los tipos de datos permitidos serán int, string, boolean, lista o arreglo y hash. Eventualmente se puede definir un tipo “void” para métodos que no retornen, si se considera necesario
* Independientemente sea en esta etapa o en el intérprete, la ejecución del programa debe verificar y reportar posibles errores de tipos, en donde privará solamente reglas de tipos simples aplicadas a operaciones entre si. Por ejemplo enteros con enteros y cadenas con cadenas.
* Los tipos de datos en parámetros en esta etapa pueden ser complicados de manejar, por lo que debe tratarse con especial atención su análisis antes de generar código.
* Recordemos que los tipos de datos son dinámicos, así que se puede tener referencia a un tipo para un identificador pero eventualmente cambiar a otro sobre la marcha.
* LA VERIFICACIÓN DE TIPOS DE DATOS PUEDE REALIZARSE TANTO EN ESTA ETAPA COMO EN LA MÁQUINA VIRTUAL.

Estructura del código fuente

Debido a las características de la máquina virtual, debe existir un punto de arranque para la ejecución del código.

Se contará con un método main al final de cada código de prueba y este método main será el punto de inicio del programa. Quiere decir por ejemplo, que si el programa es un factorial, habrá el código para dicho factorial y al final un método main que debe llamar al factorial con los parámetros reales que se quieran.

CUALQUIER OTRO DETALLE DE IMPLEMENTACIÓN SERÁ ACLARADO OPORTUNAMENTE EN EL TRANSUCRSO DE LA IMPLEMENTACIÓN DEL PROGRAMA.

**Documentación:**

La documentación será un video que describa lo siguiente:

* Soluciones e implementación sobre todo de las decisiones tomadas.
* Resultados obtenidos donde se detalle el alcance general logrado en términos de código funcional.
* Manual de pruebas donde se muestren los resultados obtenidos y la ejecución del código

**Aspectos Administrativos:**

* La tarea se desarrollará en grupos de máximo dos personas
* La fecha de entrega será el **Lunes 14 de Diciembre** del 2020 antes de las **10:00** de la noche, en el **TEC\_DIGITAL**.
* Cualquier intento de plagio, copias totales o parciales de otras personas o de Internet, serán castigados con nota de 0.