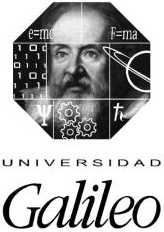
|  |  |
| --- | --- |
| Universidad Galileo | Guatemala, 21 de Julio de 2013 |
| Facultad: FISICC | **Alumnos: Ivan Somarriba; Jorge González** |
| Curso: CC4 | **Carnet: 1200 2015; 12002034** |
| Sección: A | **Día Entrega: 22/07/13 – 23:55pm** |



L

**Proyecto-Fase0**

## Clarificaciones

En si lo que tuvimos que pedir que nos aclararan era si había que implementar las clases de tipo Scan, Parse, Ast, Irt, Semantic y Codegen, las cuales solo teníamos que crear la clase y su constructor ya que después conforme avancemos en el desarrollo del proyecto íbamos a tener que implementar cada una de ellas.

Otra cosa que tuvimos que suponer que teníamos que hacer era el tener que escribir en el archivo Output y que tipo de archivo iba a ser, también si cuando pusieran “java Compiler o Filename” teníamos que crear el archivo de output con el mismo nombre que el de Input y escribir en el archivo Output todo lo que se nos especificó en las instrucciones.

## Diseño

En si para esta fase del proyecto no tuvimos que diseñar tanto lo que era el proyecto sino que teníamos que seguir las instrucciones que se nos daban para el poder armar el esqueleto de lo que se ira fomentando el compilador, si tuvimos que empezar a darle algún diseño a nuestro proyecto con nuestras propias ideas y algunas soluciones que teníamos que implementar en alguna parte de esta fase, como ejemplo podría ser el cómo manejar algunos errores que se nos presentaron a la hora de que el usuario comenta algún error al ingresar las opciones que se le dan para poder utilizar el compilador.

Pero en si para esta parte del proyecto no tuvimos gran oportunidad de tener que pensar en un diseño concreto ya que nos daban más o menos la idea de cómo se quería la infraestructura de la Fase 0.

## Descripción de Diseño

Por el momento el diseño está en base a lo que se nos pidió que realizáramos, lo único que tal vez necesito creatividad de nosotros y el tener que pensar seria que hacer con algún manejo de errores que tal vez el usuario cometería. Una cosa es que cuando el usuario en la línea de comandos escriba “java Compiler –h –o outputfile –target codegen Filename”, nosotros decidimos que al poner –h él no tenía idea de lo que está haciendo y solo le tira la documentación de ayuda e ignora los demás comandos para cuando el termine de ver la ayuda sepa utilizar correctamente cada comando y opción que tiene el compilador.

Otra cosa que tuvimos que decidir es cuando el usuario escriba en la línea de comandos “java Compiler –o output –targer semantic – targer scan Filename” es que tomaría en cuenta solo los últimos de cada tipo de opción ya que podría ser que el usuario comenta algún error en la línea de comandos y quiera que se tome en cuenta el ultimo que él escribió en la lista de opciones, también porque era una de las especificaciones adicionales de las instrucciones.

## Errores Posibles

Nosotros en lo general pensamos que nuestro programa funciona al 100% y logra realizar todo lo que se le puede pedir por el momento que realice. Talvez puede que no nos demos cuanta de algunos posibles errores ahorita que otras personas miren como un error pero en lo general pensamos que funciona todo lo que se nos pidió que realizara.