|  |  |
| --- | --- |
| Día | Resumen |
| Lunes 10 Marzo | Se tomó como base la tarea de Little Duck para codificar el léxico de nuestro lenguaje. Además, se creó un repositorio en Github para mantener una sincronía de archivos y para llevar un control de versiones. El repositorio está en:  <https://github.com/zavalza/Vispi>. Iván y Paul se dedicaron a cambiar los tokens de la tarea por los presentados en la propuesta de la semana pasada. La tarea involucró la traducción de los tokens de español a inglés y la adición de tokens presentes en lenguaje C y Python. Se utilizó la máquina virtual de Koding, para editar los archivos al mismo tiempo. |
| Martes 11 Marzo | Se tomó la gramática de Little Duck y se declararon todas las variables sintácticas de gramática formal de acuerdo a lo presentado en la propuesta de proyecto. Se actualizó el repositorio en github. Queda pendiente probar que todo funcione antes de dar por terminada la gramática. Juan Pablo. |
| Jueves 13 Marzo | Juan Pablo hizo diagramas con herramienta web |
| Miércoles 19 Marzo | Paul comienza la estructura para analizar la semántica, se investiga sobre la implementación de diccionarios en Python. |
| Jueves 20 Marzo | Juan Pablo y Paul terminaron la implementación del cubo semántico y del directorio de procedimientos. Se probó con solo un archivo de test, pero el comportamiento parece ser el adecuado. Faltan algunas validaciones que quedaron comentadas en el parser. También resultará conveniente analizar el cubo para ver la facilidad de implementación; por ejemplo, en las operaciones que involucran imágenes. |
| Jueves 27 Marzo | Se realizaron cuádruplos de expresiones simples como operaciones aritméticas. Los cuádruplos utilizan direcciones de variable y parecen no contener errores. Faltan cuádruplos de expresiones condicionales y de rutinas. |
| Jueves 3 Abril | Se desarrollaron los cuádruplos para estatutos condicionales. Se encontró un problema con el do while en la gramática, que hace que no se pueda poner un tab antes del while. Se empezaron a desarrollar los cuádruplos de las funciones, pero quedaron dudas que serán resueltas en clase cuando regrese la maestra. |
| Jueves 10 Abril | Se completó la generación de cuádruplos para las funciones y se generó el archivo .obj para que sea leído por la máquina virtual. Se añadieron funciones en la gramática para comprobar la identación del código fuente y en las primeras pruebas parece funcionar correctamente. Seguimos probando con el archivo test3.txt utilizando: python vispy.py test3.txt Mañana se espera desarrollar código en C (Máquina Virtual) que interactúe con lo generado en python. |
| Jueves 24 Abril | Se trabajó con la máquina virtual. Se decidió que se realizaría en Python por la facilidad del uso de las variables. También se acordó la manera en que la VM empezaría a estructurar el código destino (.cpp). Por ejemplo, el hecho de que las librerías estarían en otro archivo que se incluiría, la estructura general el programa en C++, etc. Del mismo modo se iniciaron pruebas para observar cómo iba quedando el código destino (se empezó a producir el código en .cpp). |
| Lunes 28  Abril | Paul trabajó con la máquina virtual. Se lograron pasar todas las variables declaradas en orden, al igual que se corrigieron algunos errores. |
| Miércoles 30  Abril | Se siguió trabajando con la máquina virtual para lograr resolver el paso de parámetros de las funciones, así como la declaración de variables locales dentro de las mismas. Se resolvieron otros errores, y se trabajó con las funciones especiales. |
| Jueves 1ro de Mayo | Se hizo un avance considerable de la máquina virtual, la cual ya genera código compilable de C++. Declaramos las variables locales y se terminaron 7 funciones especiales del lenguaje, mismas que faltan integrar a los cuádruplos además de corregir algunos bugs. |