

MANUAL TECNICO CH-MAQUINA.

1. Introducción

¿Qué es el ch-maquina?

Primero que todo el ch-maquina es un simulación de un sistema operativo arcaico, en el cual lo que se debe tener en cuenta son 2 aspectos importantes como lo son la administración de la memoria del sistema, y la ejecución y administración de programas.

1.1. Administración de memoria.

La administración de memoria consiste en realiza un vector de 99999 espacios , el cual se tiene que mostrar su funcionamiento en pantalla, aspectos como que contiene en el momento y que se está cargando(Por motivos de facilidad se mostrara en pantalla las primeras 100 posiciones del vector), además se requiere simular el sistema operativo en la memoria, el cual requiere de un espacio en ella, por ende se reservara la cantidad de $10 \cdot z + 9$ posiciones, en donde Z es el último dígito de nuestra cedula y el cual se alojara en las primeros espacios de memoria.

1.2. Ejecución y administración de programas.

La ejecución y administración de programas es la manera como representamos la acción que hace el sistema para mostrarnos los programas como cotidianamente los conoces, en el cual realizamos un programa en X lenguaje de programación y el "automagicamente" realiza todas las operaciones para transformarlo como una aplicación ejecutada y listo para utilizar.

Como nuestro sistema operativo es un poco arcaico y no contiene la misma cantidad de herramientas y opciones que los convencionales, solo aceptara un lenguaje de programación, el cual es llamado ch-Código y corresponde un lenguaje de programación muy sencillo y similar a el assembler, del cual hablaremos más adelante.

El proceso de ejecución y administración de programas se partirá en varios sub- procesos con los cuales entenderemos un poco más como se realizar dicho proceso y los cuales son verificación sintaxis de los .ch (los .ch son los archivos que alojan el ch-Código), cargar

los .ch a la memoria e interpretación y ejecución del ch-Código

1.2.1 Verificación sintaxis de los .ch

Como en todo lenguaje de programación, a la hora de programar, existen infinidad de errores humanos, los cuales pueden ser desde una no declaración de variable hasta un ciclo mal estandarizado, por ende se requiere que todos estos errores de mecanográfica de los cuales ninguno de nosotros estamos exceptos sean corregidos y más verificar cada una de las líneas de código para asegurar que se utilice de manera correcta las palabras reservadas y se declare de manera óptima cada una de las variable.

1.2.2 cargar los .ch a la memoria.

Ya que se ha realizado toda la verificación de sintaxis, se procederá a cargar cada una de las líneas de código a la memoria, las cuales serán cargadas en líneas continuas empezando en los espacios libres después de los reservados para el sistema operativo.

1.2.3 Interpretación y ejecución del ch-Código.

A la hora de realizar la interpretación y ejecución se debe estar seguro que se ha verificado la sintaxis del ch-Código y se ha cargado todo el ch-Código a memoria, por ende cada uno de estos procesos debe ser secuencial y directamente condicionados.

La interpretación se refiere al manejo de cada una de las líneas en ch-Código y traducirlas a un lenguaje conocido, de esta manera se simulara el proceso normal que lleva a cabo cada uno de los programas de nuestro sistema operativo a la hora de llevarlo a lenguaje máquina.

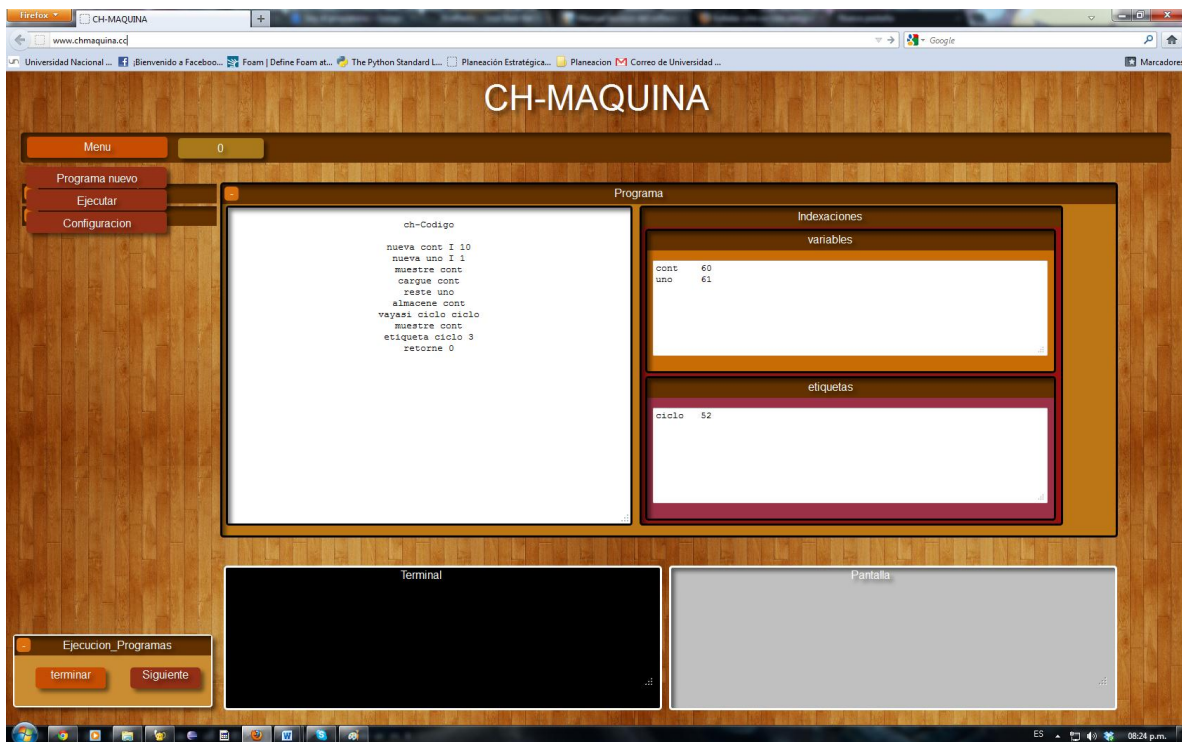
La ejecución del programa es bastante sencilla si se han seguido todos los pasos anteriores ya que se puede asegurar que todo lo que se corre por nuestro ch-Maquina está debidamente estandarizado y si podrá realizar todas las laboras para lo que ha sido realizado nuestros ch-Programas.

2. Requerimientos del ch-maquina.

Esta aplicación debe ser ejecutada en MOZILLA FIREFOX 3 en adelante ya que la administración de archivos aun no es compatibles con otros exploradores.

3. Apertura del sistema.

Ingresar a la link www.chmaquina.co.cc



4. Especificaciones técnicas.

4.1 Lenguaje Utilizado:

- JavaScript

Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador...⁴ aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

- CSS

El nombre **hojas de estilo en cascada** viene del inglés *Cascading Style Sheets*, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML² (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la *estructura* de un documento de su *presentación*.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>".

- HTML

Siglas de **HyperText Markup Language** («lenguaje de marcado de hipertexto»), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

HTML también es usado para referirse al contenido del [tipo de MIME](#) text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del [XML](#) (como [XHTML](#) 1.0 y posteriores) o en forma descendida directamente de [SGML](#) (como HTML 4.01 y anteriores).

4.2 Framework.

- **jQuery**

Es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC.

Ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

4.3 Clases.

- **CHMaquina()**

La aplicación ch-maquina esta manejada desde esta clase, ya que la comunicación entre clase, no existe.

La simulación del procesador carga los archivos, depura las líneas y ejecuta el programa.

- **Memoria()**

Es todo lo relacionado con la administración de la memoria, insertar programas, entregar líneas para ejecución entre otras funciones que son pedidas por la clase CHMaquina

- **Depurador()**

Es la primera parte para una aplicación, ya que si no se encuentra ningún problema en el programa a insertar se podrá ejecutar, esta clase interactúa con trato_linea

- **Trato_lineas()**

No es realmente una clase sino un módulo donde se encuentran todas las funciones relacionadas con las palabras reservadas del ch-maquina este es llamado por la clase depurador.