****

**UNIVERSIDAD HISPANOAMERICANA.**

**Curso: Programación II.**

**Día: Miércoles en la Tarde.**

**Carrera: Ingeniería En Sistemas.**

**Tema: Tarea1**

**Estudiante:**

**González Solano Rigoberto.**

Lenguajes y sus diferencias:

C.

-Desarrollado en el año [1972](http://www.ecured.cu/1972) por [Dennis Ritchie](http://www.ecured.cu/Dennis_Ritchie) para [UNIX](http://www.ecured.cu/UNIX) un sistema operativo multiplataforma.

Es del tipo lenguaje estructurado como son [Pascal](http://www.ecured.cu/Pascal), [Fortran](http://www.ecured.cu/Fortran), [Basic](http://www.ecured.cu/index.php?title=Basic&action=edit&redlink=1).

-Sus instrucciones son muy parecidas a otros lenguajes incluyendo sentencias como if, else, for, do y while.

-Tiene una gran facilidad para escribir código compacto y sencillo a su misma vez.

-No tenemos procedimientos como en otros lenguajes solamente tenemos funciones los procedimientos los simula y esta terminante mente prohibido escribir funciones, procedimientos y los comandos en mayúscula todo se escribe en minúsculas (a no ser las constantes J).

C++

-El lenguaje C++ se comenzó a desarrollar en 1980. Su autor fue [Bjarne Stroustrup](http://www.ecured.cu/Bjarne_Stroustrup" \o "Bjarne Stroustrup),

-Es un lenguaje imperativo orientado a objetos derivado del C.

-Sigue muy ligado al hardware subyacente, manteniendo una considerable potencia para programación a bajo nivel, pero se la han añadido elementos que le permiten también un estilo de programación con alto nivel de abstracción.

-C++ es una extensión del C clásico

-Es un lenguaje de propósito general basado en el C, al que se han añadido nuevos tipos de datos, clases, plantillas, mecanismo de excepciones, sistema de espacios de nombres, funciones inline, sobrecarga de [operadores](http://www.ecured.cu/Operadores), referencias, operadores para manejo de memoria persistente, y algunas utilidades adicionales de librería.

C#

-Es un nuevo lenguaje de programación diseñado para crear un amplio número de aplicaciones empresariales que se ejecutan en .NET Framework.

 -Es sencillo, moderno, proporciona seguridad de tipos y está orientado a objetos.

-El código creado mediante C# se compila como código administrado, lo cual significa que se beneficia de los servicios de Common Language Runtime.

-Creado por el danés Anders Hejlsberg.

-Este nuevo lenguaje se quiso mejorar con respecto de los dos lenguajes anteriores de los que deriva el C, y el C++.

Git

Emplea una serie de archivos de configuración. El primero de ellos es el archivo '/etc/gitconfig', que contiene valores para todos y cada uno de los usuarios en el sistema y para todos sus repositorios. Con la opción '--system' del comando 'git config', puedes leer y escribir de/a este archivo.

El segundo es el archivo '~/.gitconfig', específico para cada usuario. Con la opción '--global', 'git config' lee y escribe en este archivo.

Y por último, Git también puede considerar valores de configuración presentes en el archivo '.git/config' de cada repositorio que estés utilizando. Estos valores se aplicarán únicamente a dicho repositorio. Cada nivel sobreescribe los valores del nivel anterior; es decir lo configurado en '.git/config' tiene primacia con respecto a lo configurado en '/etc/gitconfig', por ejemplo. También puedes ajustar estas configuraciones manualmente, editando directamente los archivos correspondientes y escribiendo en ellos con la sintaxis correspondiente; pero suele ser más sencillo hacerlo siempre a través del comando 'git config'.

**1. Git help** comando ó **git** comando :

    Muestra la ayuda para ese comando.

**2.Git help** comando ó **git** comando :

    Muestra la ayuda para ese comando.

3. **Git add:**

    Adiciona un archivo o un directorio de manera recursiva.

4. **Git rm :**

    Remueve un archivo o directorio del árbol de trabajo.

Fuerza la eliminación de un archivo del repositorio.

5. **Git mv:**

    Mueve el archivo o directorio a una nueva ruta.

    Sobre-escribe los archivos existentes en la ruta destino.

6. **Git status:**

    Imprime un reporte del estado actual del árbol de trabajo local.

7. **Git add:**

    Selecciona el archivo para que sea incluido en el próximo commit.

8. **Git commit:**

    Realiza el commit de los archivos que han sido registrados (con git-add).

9. **Git log :**

    Muestra el log del commit, opcionalmente de la ruta especifica.

10. . **Git pull:**

    Descarga y guarda los cambios realizados desde un repositorio remoto.

11. **Git push :**

    Guarda los cambios en un repositorio remoto.

Github

Es una plataforma para alojar proyectos.