Diferencias entre C, C++ y C#

Prof: Jorge Davian Aguilar

Universidad Hispanoamericana Programacion-2

Jose Carlos Acuña Rojas

**Diferencias Entre C, C++ y C#.**

**Lenguaje C**

También conocido como “Lenguaje de programación de sistemas” desarrollado en el año 1972 por Dennis Ritchie para UNIX, un sistema operativo multiplataforma. El lenguaje C es un tipo lenguaje estructurado como: Pascal, Fortran, Basic. Sus instrucciones son muy parecidas a otros lenguajes incluyendo algunas como: if, else, for, do y while, etc. Aunque C es un lenguaje de alto nivel ya que es estructurado y posee sentencias y funciones que simplifican su funcionamiento, tenemos la posibilidad de programar a bajo nivel como en el Ensamblador tocando los registros, memoria etc.

**Lenguaje C++**

C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

**Lenguaje C#**

C# es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET, similar al de Java, aunque incluye mejoras derivadas de otros lenguajes.

Por supuesto, esta historia sintetizada de C# se enfoca sólo en la rama evolutiva que corresponde a C, C++ y Java. En 1964 comenzó otra notable ramificación con la invención de un lenguaje de programación para principiantes denominado BASIC, el cual evolucionó hasta convertirse en lo que se conoce hoy como Visual Basic, y que también forma parte de “punto net”.

**Diferencias**

C, se trata de un lenguaje débilmente tipificado de medio nivel, pero con muchas características de bajo nivel. Dispone de las estructuras típicas de los lenguajes de alto nivel pero, a su vez, dispone de construcciones del lenguaje que permiten un control a muy bajo nivel. Los compiladores suelen ofrecer extensiones al lenguaje que posibilitan mezclar código en ensamblador con código C o acceder directamente a memoria o dispositivos periféricos.

La programación en C tiene una gran facilidad para escribir código compacto y sencillo a su misma vez. En el lenguaje C no tenemos procedimientos como en otros lenguajes solamente tenemos funciones, los procedimientos los simula y está terminantemente prohibido escribir funciones los procedimientos y los comandos en mayúscula, todo se escribe en minúsculas de otra manera generara errores; una vez escrito se debe pasar a compilar el archivo; los archivos tienen 2 Extensiones el archivo.C que es el archivo a compilar el que contiene todas los procedimientos funciones y código de nuestro programa y archivo.h que son las librerías que contienen las funciones de nuestro programa. El compilador genera Archivos con extensión .EXE

C es un lenguaje libre estandarizado por ISO MUY PEQUEÑO que admite programación estructurada. Útil en programación de microchips, sistemas operativos, drivers y programación web.

Una particularidad del C++ es la posibilidad de redefinir los operadores ya que se puede utilizar la sobrecarga de operadores, y el poder crear nuevos tipos que se comporten como tipos fundamentales.

C++ es un lenguaje libre estandarizado por ISO MUY GRANDE, que admite: Programación estructurada, la totalidad de la POO, programación genérica (plantillas, STL, conceptos de contenedores, estructuras de datos genéricas, <b>metaprogramación</b>... es una programación que no dispone ningún otro lenguaje mayoritario por el momento) y otras características más avanzadas como los punteros inteligentes, programación lambda, programación "física"... También se usa como programación por eventos (MFC, Qt, Gtk y otras GUIs). Se usa para cualquier cosa, desde sistemas operativos hasta los juegos 3D de última generación pasando por servidores, pasando por las típicas aplicaciones de escritorio o un Office o OpenOffice, un reproductor WinAMP y Windows Media Player, un explorador web (Internet Explorer, Mozilla, Firefox...), un cliente eMule, un cliente Torrent.

C# es un invento de Microsoft (lenguaje propietario) que mezcla las características básicas de C++ excepto las avanzadas simplificándolas al estilo Java y ofreciendo un framework. El problema es que es .Net, y deja de ser código nativo/portable. Eso sí, el framework provee bastante facilidad de programación de tareas comunes, al igual que Java. Por ello se genera el debate ¿Java o C#? Su funcionalidad viene a ser parecida. .Net es más nativo y Java más virtual.

**Git**

Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end como Cogito o StGIT. Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de programación del núcleo Linux.

**Comandos de Git**

git fetch: Descarga los cambios realizados en el repositorio remoto.

git merge <nombre\_rama>: Impacta en la rama en la que te encuentras parado, los cambios realizados en la rama “nombre\_rama”.

git pull: Unifica los comandos fetch y merge en un único comando.

git commit -am "<mensaje>": Confirma los cambios realizados. El “mensaje” generalmente se usa para asociar al commit una breve descripción de los cambios realizados.

git push origin <nombre\_rama>: Sube la rama “nombre\_rama” al servidor remoto.

git status: Muestra el estado actual de la rama, como los cambios que hay sin commitear.

git add <nombre\_archivo>: Comienza a trackear el archivo “nombre\_archivo”.

git checkout -b <nombre\_rama\_nueva>: Crea una rama a partir de la que te encuentres parado con el nombre “nombre\_rama\_nueva”, y luego salta sobre la rama nueva, por lo que quedas parado en ésta última.

git checkout -t origin/<nombre\_rama>: Si existe una rama remota de nombre “nombre\_rama”, al ejecutar este comando se crea una rama local con el nombre “nombre\_rama” para hacer un seguimiento de la rama remota con el mismo nombre.

git branch: Lista todas las ramas locales.

git branch -a: Lista todas las ramas locales y remotas.

git branch -d <nombre\_rama>: Elimina la rama local con el nombre “nombre\_rama”.

git push origin :<nombre\_rama>: Elimina la rama remote con el nombre “nombre\_rama”.

git remote prune origin: Actualiza tu repositorio remoto en caso que algún otro desarrollador haya eliminado alguna rama remota.

git reset --hard HEAD: Elimina los cambios realizados que aún no se hayan hecho commit.

git revert <hash\_commit>: Revierte el commit realizado, identificado por el “hash\_commit”.

**GitHub**

GitHub es una plataforma de desarrollo de software de manera colaborativa para alojar proyectos utilizando el sistema de control de versiones Git. GitHub permite alojar en un repositorio propio de código y brinda herramientas muy útiles para el trabajo en equipo, dentro de un proyecto. Realizar un fork es simplemente clonar un repositorio ajeno (genera una copia en su cuenta), para eliminar algún bug o modificar cosas de él. Una vez realizadas las modificaciones, se puede enviar un pull al dueño del proyecto. Éste podrá analizar los cambios que se han realizado fácilmente, y si considera interesante su contribución, adjuntarlo con el repositorio original.

**Como se puede usar**

En nuestra especialidad “Programación”, fuimos aprendiendo cosas y creando programas de código abierto, fomentando el software libre; es por eso que presentamos esta gran herramienta enfocada al crecimiento de proyectos comunitarios y libres. En esta página podremos crear una cuenta gratuita y comenzar a subir repositorios de código (o crearlos desde 0), para que con la ayuda de todos, ese proyecto mejore; así como también fortalecer los proyectos de los demás para crecer como grupo.