

Bueno he aquí el trigésimo séptimo tuto de HDC. Espero todos sientan agradable el curso y saben que cualquier sugerencia, idea o cambio que les gustara hacer, manden mail o comenten en el curso que muchas veces se me ocurren cosas pero que no sé si sería buen cambio o demasiado arriesgado. Para mí, esto del poder hablar con ustedes y tener buena relación con muchos de los participantes es muy agradable. :D

Aquí estamos, aprendiendo a programar o queriendo ser parte de este mundo. Hoy veremos cómo podemos instalar el compilador de este famoso y potente lenguaje, C.

Introducción a C:

Este lenguaje fue creado en épocas de antaño -1972- por un tipo llamado **Dennis Ritchie**. Lo bueno de este código es que es de **nivel intermedio** pero más llegando a **bajo nivel** para poder manejar bits. También hay lugares donde podemos mezclarlo con **Assembler** -un lenguaje de nivel casi tan básico como escribir con 1 y 0-. Es muy flexible como para poder usarse en la creación de aplicaciones o de sistemas operativos. Es un lenguaje **estructurado** (ya vimos lo que era, digamos que sigue las instrucciones paso a paso de arriba a abajo), pero se pueden usar algunas **rupturas** sobre éste que siendo más experimentados, conoceremos.

Tiene algunas carencias pero imaginen que muchos lenguajes son **hijos del C**.

Pero bueno, ahora lo primero que tengo que decir es que vamos a instalar el compilador en W7. Es necesario decir que lo voy a hacer en **Windows**, porque hay veces que voy a enseñar ciertos comandos que en Linux son distintos y **no funcionarían**. Por eso comento que vamos a usar W7 (los que tengan hecha la máquina virtual, aprovéchenla para cosas como éstas).

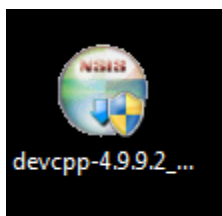
El compilador que utilizaremos será "**Dev-C++**". Nos servirá para más adelante cuando veamos C++ también, porque es compatible con ambos lenguajes:D

Lo descargamos de la página oficial (parece un tanto vieja pero prometo que no defrauda)

<http://www.bloodshed.net/dev/devcpp.html>

Bajamos el que diga: "**Dev-C++ 5.0 beta 9.2 (4.9.9.2) (9.0 MB) with Mingw/GCC 3.4.2**".

Éste será el instalador.

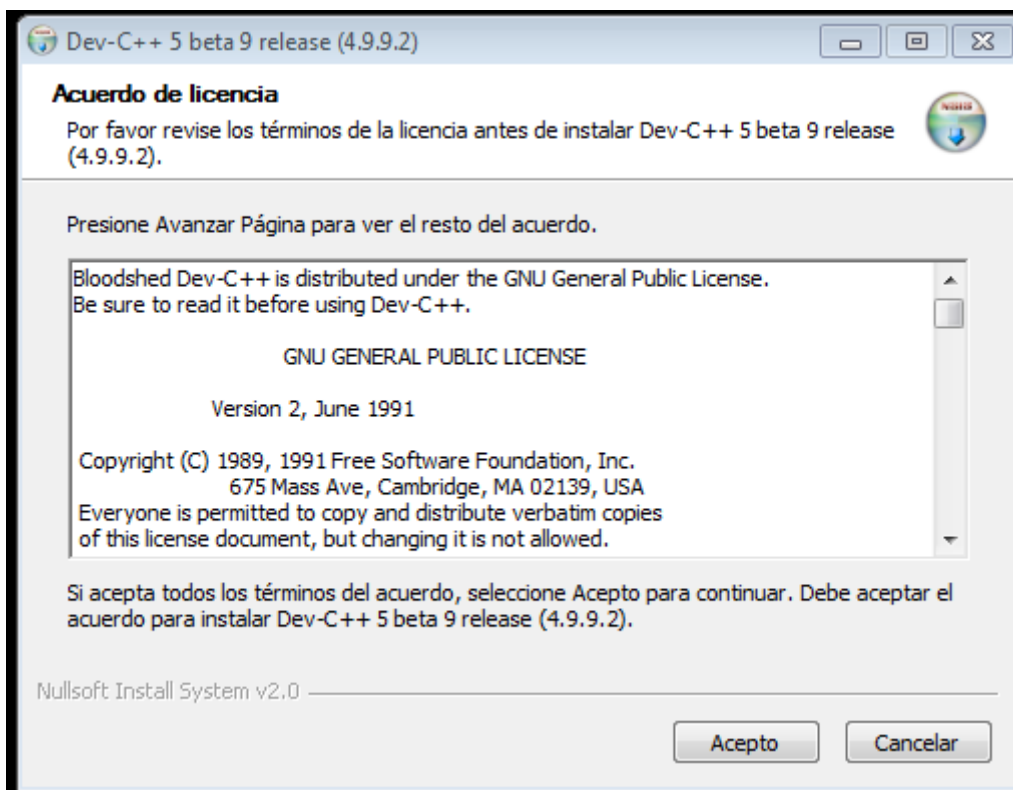


Les dará una ventanita de que no instalemos el programa si ya tenemos una versión antigua en el sistema. En ese caso deberíamos desinstalar la versión anterior.

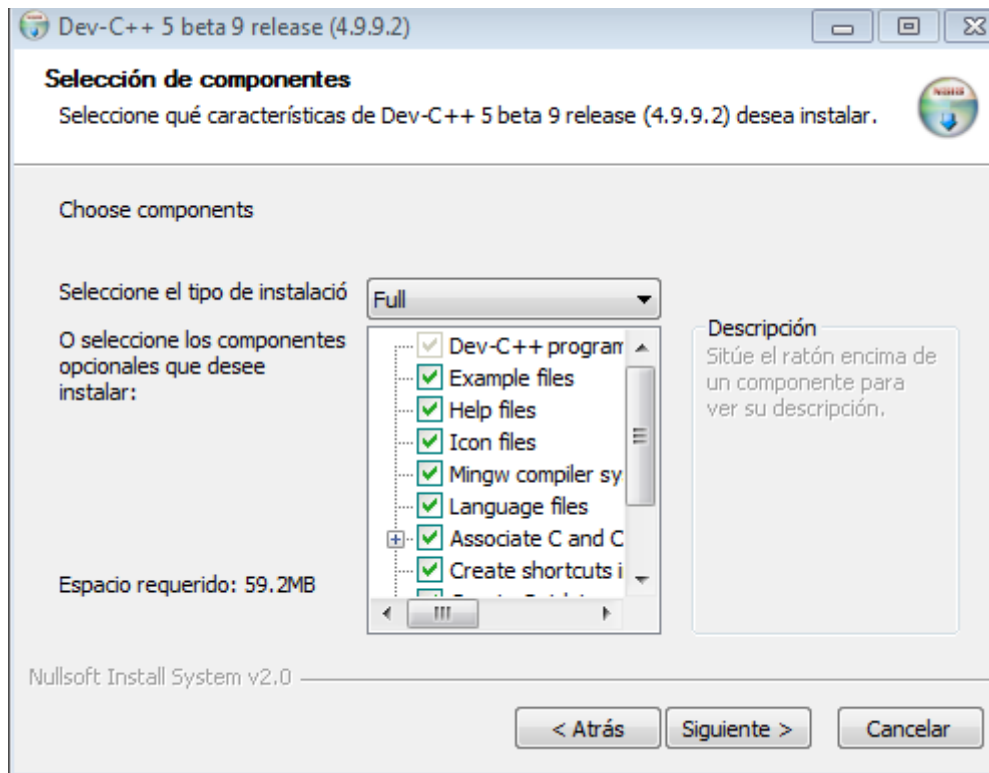
Y escogemos el idioma en el que queremos que esté el programa.



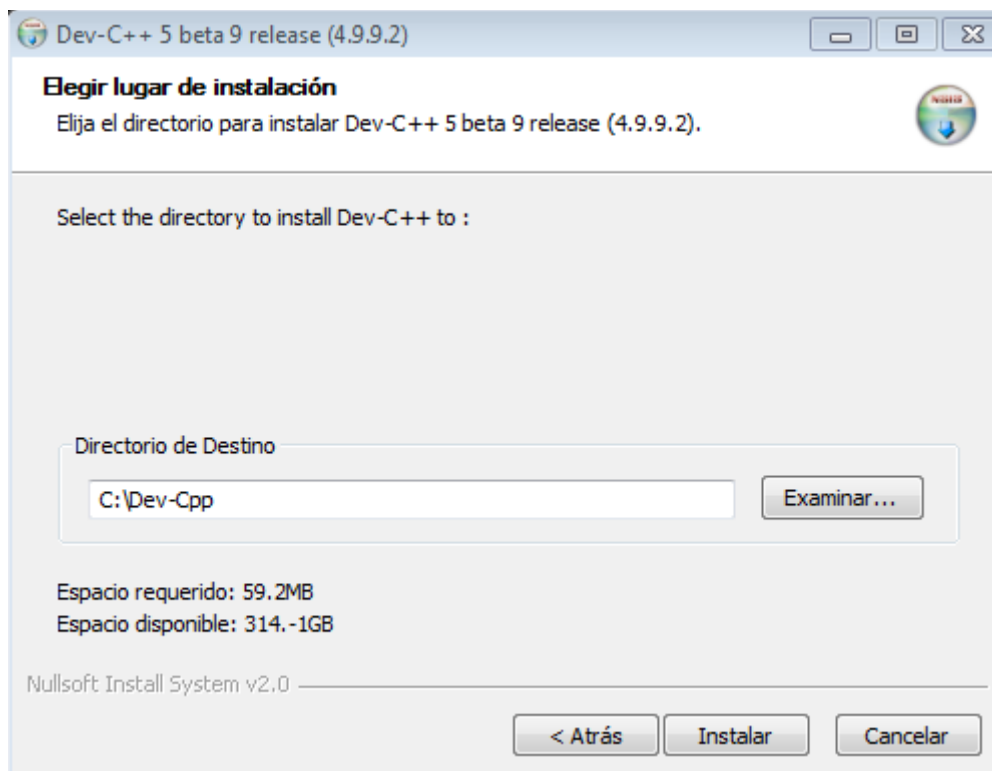
Como siempre, el contrato que debemos leer y aceptar -siempre es mejor leerlo que dejarlo pasar, pero yo ya lo leí anteriormente-.



Seleccionamos el tipo “full” de instalación.

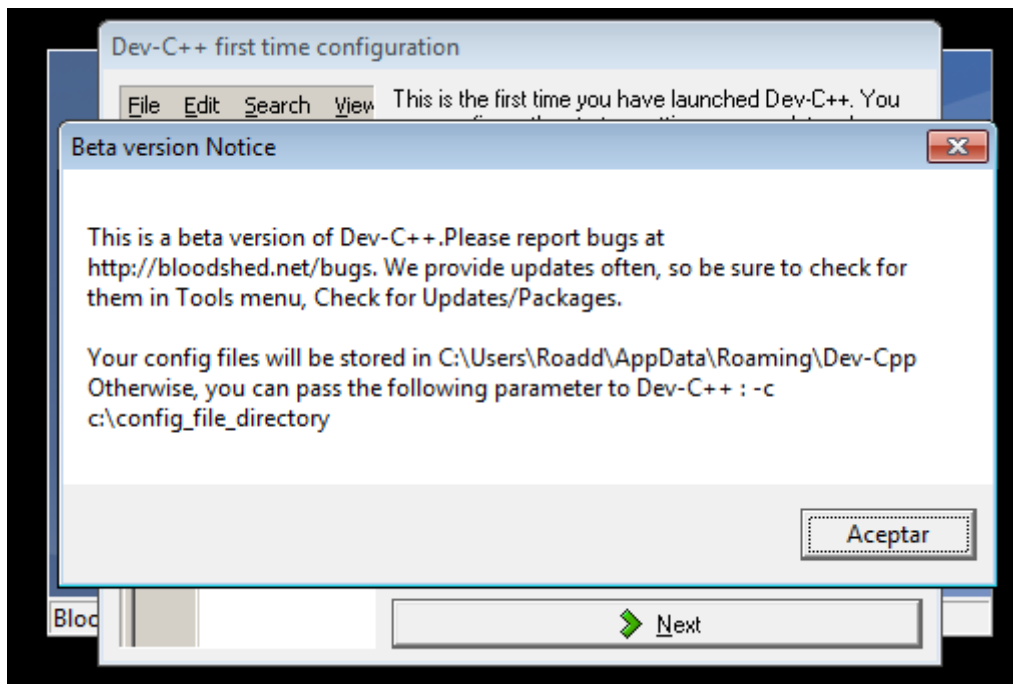


Elegimos el lugar donde se instalará el programa (es conveniente dejarlo como está por default). Y elegimos la obvia opción de “Instalar”.

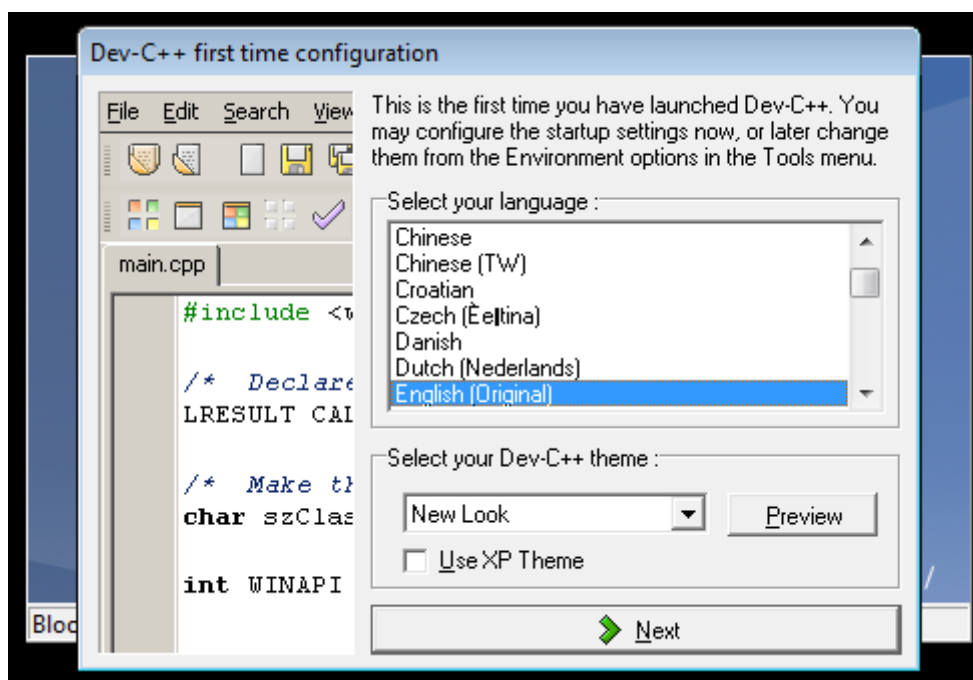


Luego preguntará si queremos instalarlo para todos los usuarios y si queremos ejecutarlo luego de terminar la instalación. Elegimos como queremos y termina:D.

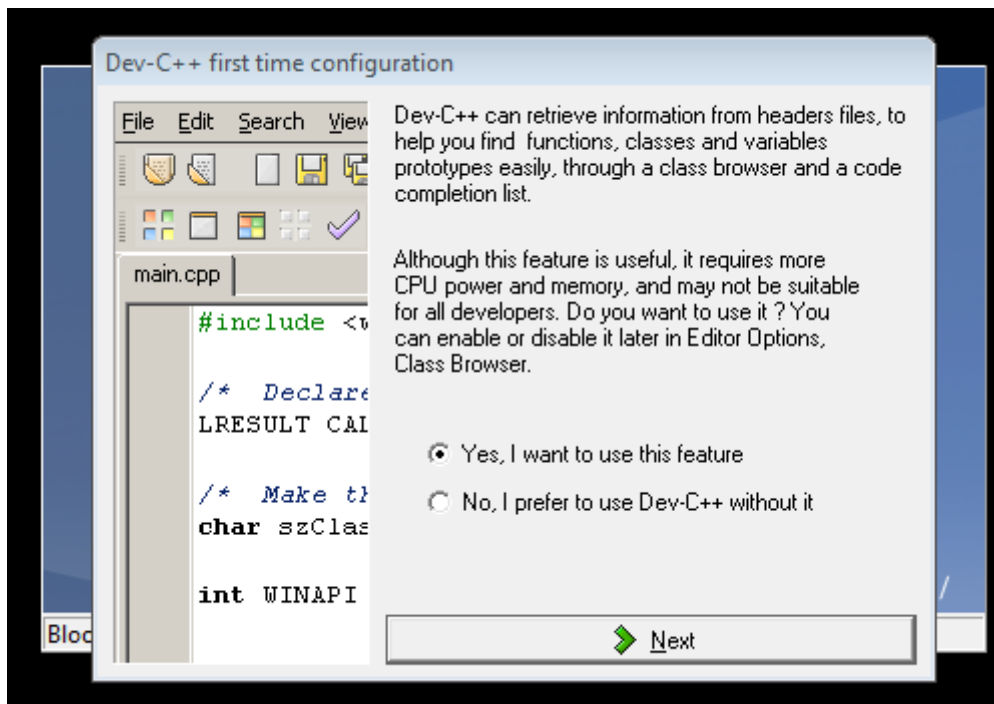
Abramos por primera vez el entorno gráfico del compilador:) (seguramente no se haya creado el acceso directo, pero en el menú de inicio debería estar). Nos encontraremos con ésto.



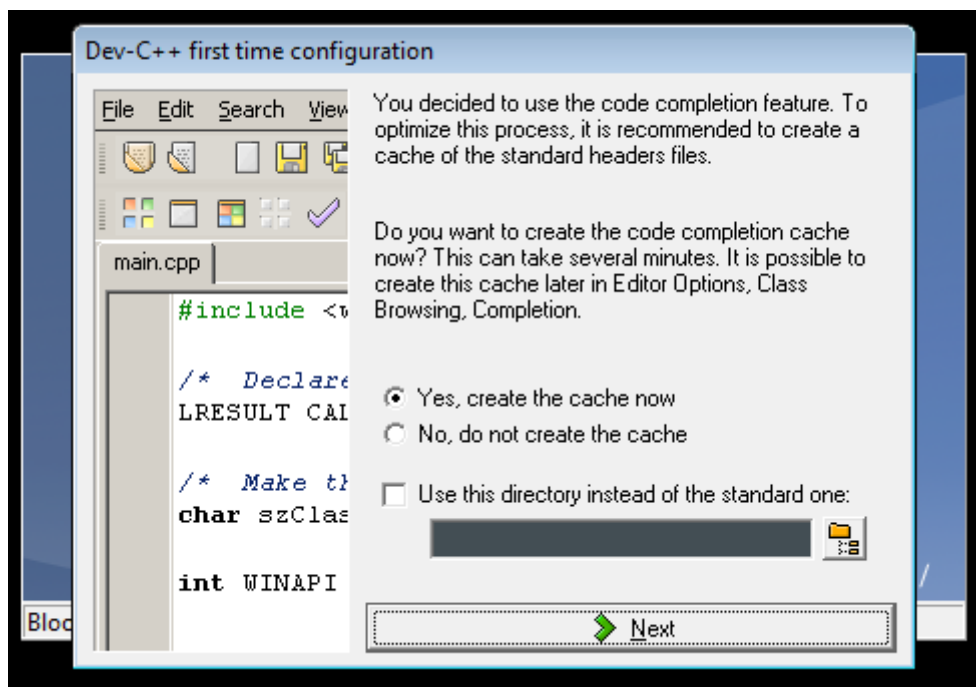
Nos avisa que es una versión beta del software, y algunas cositas más. Le damos a aceptar.



Ahora nos pide el “theme” del software y el lenguaje. Yo lo dejaré en inglés, porque es universal y así no tengo que pensar la mitad de las cosas en español y la otra mitad en inglés.

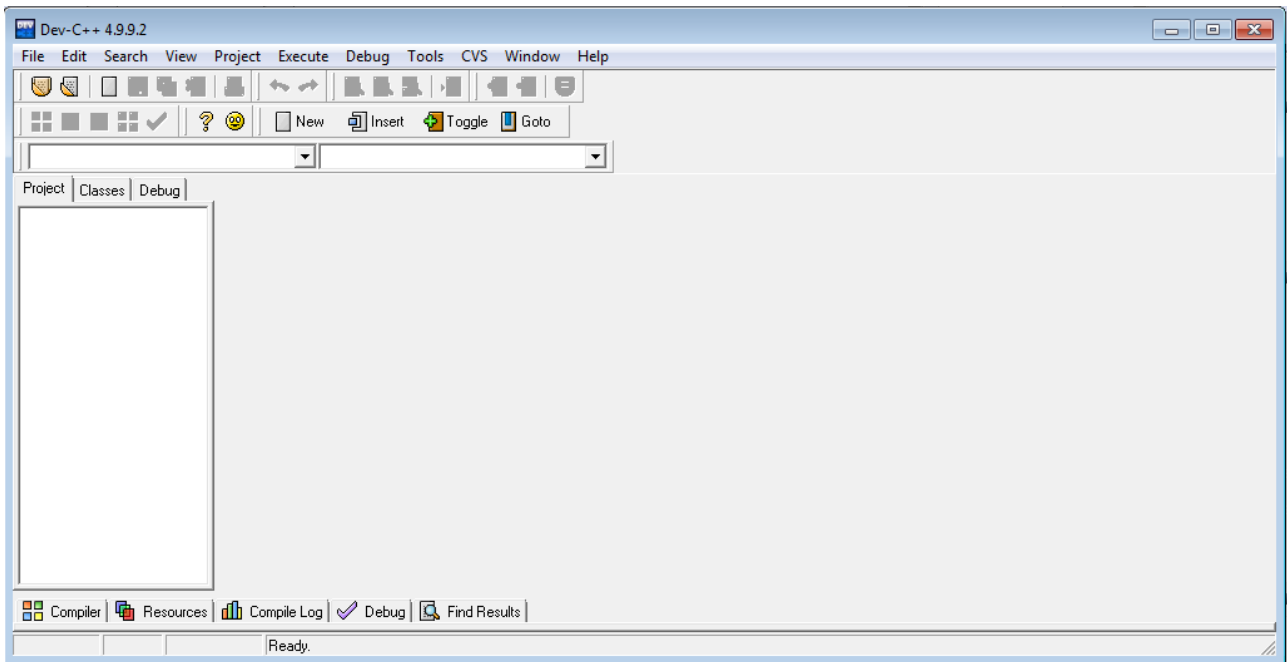


En ésta dejamos como está.

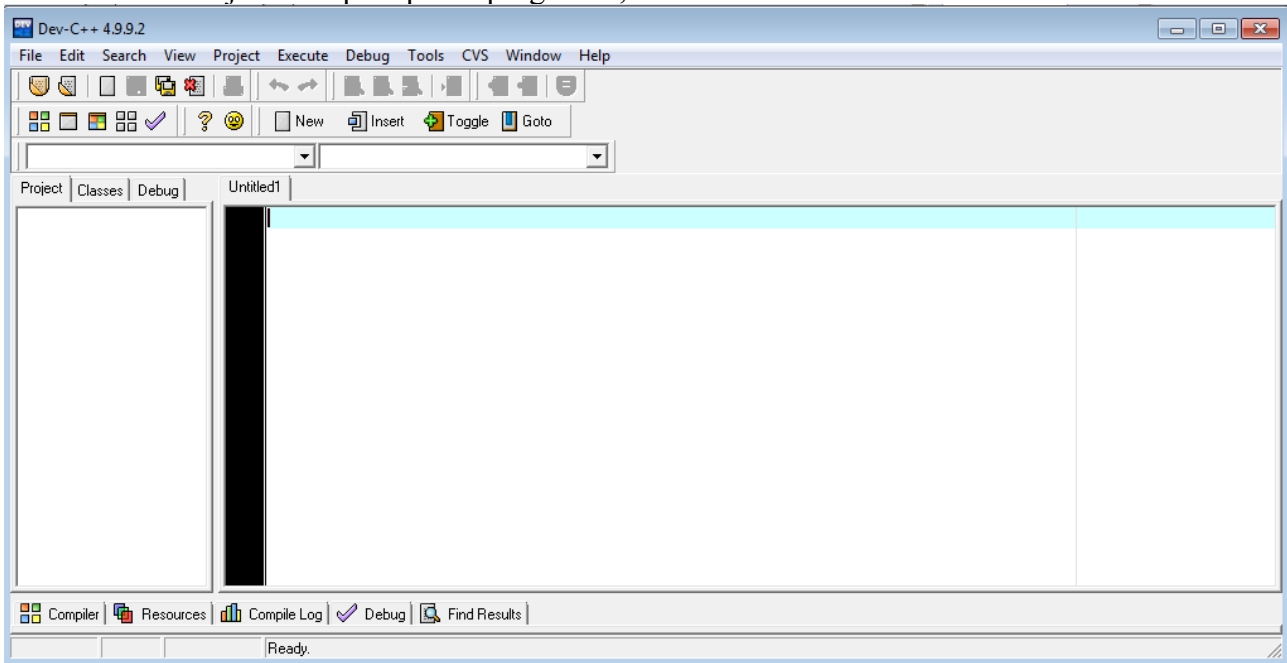


Seguimos dejando como está. Queremos que nos cree el cache ahora. Tarda un poquito, pero sean pacientes.

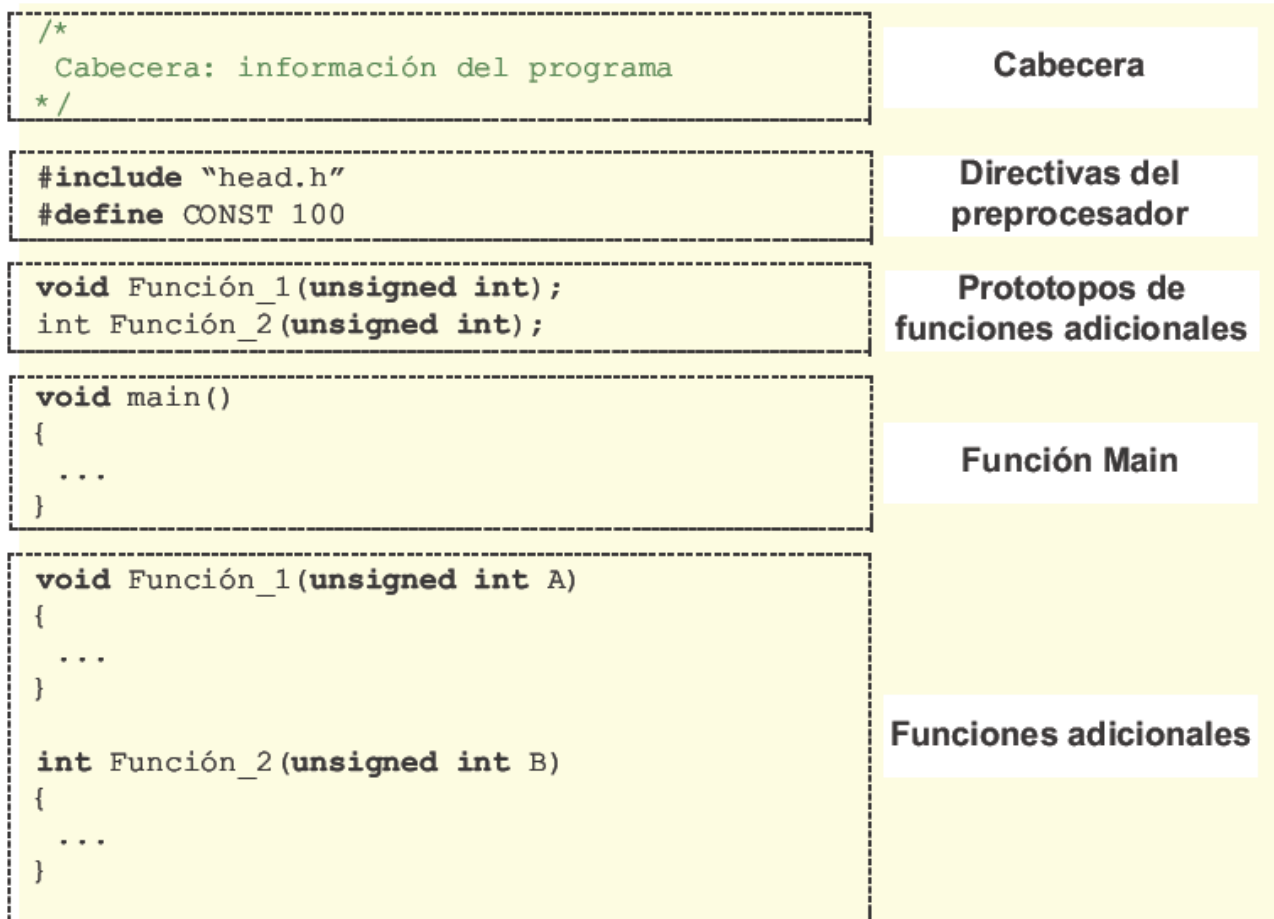
Luego cuando está por iniciar nos da un tip y nos dice si queremos que sigan habiendo al inicio. Yo le digo que no.^^! Y aquí tenemos nuestro entorno de programación. Parece vacío, luego lo llenaremos de cosas.



Para crear una hoja nueva para poder programar, se va a File → New → source file.



El C tiene reglas como veníamos viendo con el pseudocódigo. Tiene una cierta estructura que debemos respetar, globalmente hablando. Veamos cuál es:



Voy a terminar el tuto explicando ésto último.

Primero la **cabecera**. Se usa para dar información del programa. Es transparente para el usuario, pero el que programa lo ve. Es decir que es una simple ayuda para que el programador sepa con qué se va a encontrar. No es obligatoria pero ayuda mucho.

Las **directivas del preprocesador** tiene 2 partes. La parte de los “#include”, que sirve para agregar librerías que tienen montones de facilidades para los programadores porque son pequeñas porciones de código ya escrito; y los “#define” con el que se pueden crear igualdades entre un nombre y un valor específico.

Luego voy a comentar que el “**main**” es el lugar donde nosotros colocaremos el código para que corra y luego podamos ejecutar.

Aunque recién estoy dando una pequeñísima introducción a C, espero que les sea algo útil. Si quieren pueden empezar a buscar información en la red y probar por ustedes mismos. **Yo ahora me toca la tarea de hacer el próximo curso:).**

Cualquier cosa pueden mandarme mail a: r0add@hotmail.com

Para donaciones, pueden hacerlo en bitcoin en la dirección siguiente:

1HqpPJbbWJ9H2hAZTmPxnVuoLkKp7RFSvw

Roadd.

Este tutorial puede ser copiado y/o compartido en cualquier lado siempre poniendo que es de mi autoría y de mis propios conocimientos.