|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | ¿Qué es GCC y GDB? | | GCC es un compilador integrado del proyecto GNU para C, C++, Objective C y Fortran; es capaz de recibir un programa fuente en cualquiera de estos lenguajes y generar un programa ejecutable binario en el lenguaje de maquina dinde se ha de correr.  GDB (Gnu Project Debugger) es una herramienta que permite entre otras cosas, correr el programa con la posibilidad de detenerlo cuando se cumple cierta condición, avanzar paso a paso, analizar que ha pasado cuando un programa se detiene o cambiar algunas cosas del programa como el valor de las variables. | |  |  | |  | | --- | |  | |  | | Diego Armando Becerra Iñiguez  Embebidos | |  |  | |  | | --- | | Comandos GCC y GDB | | Comandos básicos usados en GCC y GDB | |  | |  | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | |  | | Comandos básicos de GDB 1.Compilar programas en C con opción -g  % gcc -g prog.c ... -o <archivo binario>  Si no se especifica -o, se genera el archivo a.out  2.gdb <archivo binario> sirve para invocar a un archivo binario.  3.help <comando> sirve para obtener ayuda  4.(gdb) b main Poner breakpoint en funciones  5.(gdb) del <nro. del break> sirve para borrar breakpoints  6.(gdb) info break Sirve para mostrar el número de breakpoints  7.(gdb) run Sirve para correr el programa  8.(gdb) where Sirve para ver el encadenamiento de funciones de la tarea actual (la que tiene la CPU en ese momento)  9.(gdb) up 🡪Si llamo a G y estamos en G, pasa a F  (gdb) down 🡪vuelve a G  Sirve para subir y bajar en la pila para ver variables de funciones intermedias   * Especifico para programas en C:   1.(gdb) s 🡪 ejecuta una instrucción. Si hay una llamada a una función, se detiene en la primera instrucción de esa función.  (gdb) n 🡪 ejecuta una instrucción. Si hay llamadas a funciones, se las ejecuta completamente sin detenerse.  2.(gdb) p x->a.d + 1 (p de print) Imprimir valores de expresiones  3.(gdb) info locals Sirve para imprimir las variables locales de la funcion examinada   * Específico para programas en Assembler:   1.(gdb) stepi -> ejecuta una instrucción de máquina. (gdb) nexti -> si es un call, ejecuta el call hasta el retorno, si no, ejecuta una instrucción de máquina.  Ambas instrucciones muestran la direccion de la siguiente instruccion a ejecutar en hexadecimal. Ej.: (gdb) 0x080483d4 in dump ()  2.(gdb) info register Sirve para mostrar el contenido de los registros  3.(gdb) x/10x 0xbffff5c8 Sirve para mostrar el contenido a partir de una direccion en Hexadecimal. | |  |  | |  | | --- | | Comandos básicos de GCC  1. .c Fuente en C 2. .C, .cc, .cpp .c++ .cp .cxx Fuente en C++;se recomienda .cpp 3. .m Fuente en Objective-C 4. .i C preprocesado 5. .ii C++ preprocesado 6. .s Fuente en lenguaje ensamblador 7. .o Código objeto 8. .h Archivo para procesador (encabezados), no suele figurar en la línea de comando de gcc 9. -E realiza solamente el preprocesamiento, enviando el resultado a la salida estándar. 10. -o archivo ,indica el nombre del archivo de salida, cualesquiera sean las etapas cumplidas. 11. -L especifica la ruta hacia el directorio donde se encuentran los archivos de biblioteca con el código objeto de las funciones referenciadas en el programa fuente.  No lleva espacio entre la L y la ruta, así: -L/usr/lib. 12. -Wall muestra todos los mensajes de error y advertencia del compilador, incluso algunos cuestionables, pero en definitiva fáciles de evitar escribiendo el código con cuidado. 13. -g incluye en el ejecutable generado la info. | | 1. -v muestra los comandos ejecutados en cada etapa de compilación y la versión del compilador. Es un informe muy detallado. | |  | |  |  | |  | | --- | | Ejemplo #include <stdio.h>  int main() {   int v = 0;     int i;   for (i = 0; i < 5; i++) {     v += i;   }   printf("Resultado: %i\n", v);   return 0;  }  $ gcc -g -o test test.c  $ gdb test     GNU gdb 5.3-debian     Copyright 2002 Free Software Foundation, Inc.     GDB is free software, covered by the GNU General Public License, and you are     welcome to change it and/or distribute copies of it under certain conditions.     Type "show copying" to see the conditions.     There is absolutely no warranty for GDB. Type "show warranty" for details.     This GDB was configured as "i386-linux"...     (gdb)  La línea "(gdb)" nos indica que el depurador está listo para empezar a procesar nuestras órdenes. El primer comando que veremos es (l)ist. Este comando muestra el código fuente del programa que estoy depurando. | |  | | GCC | |