

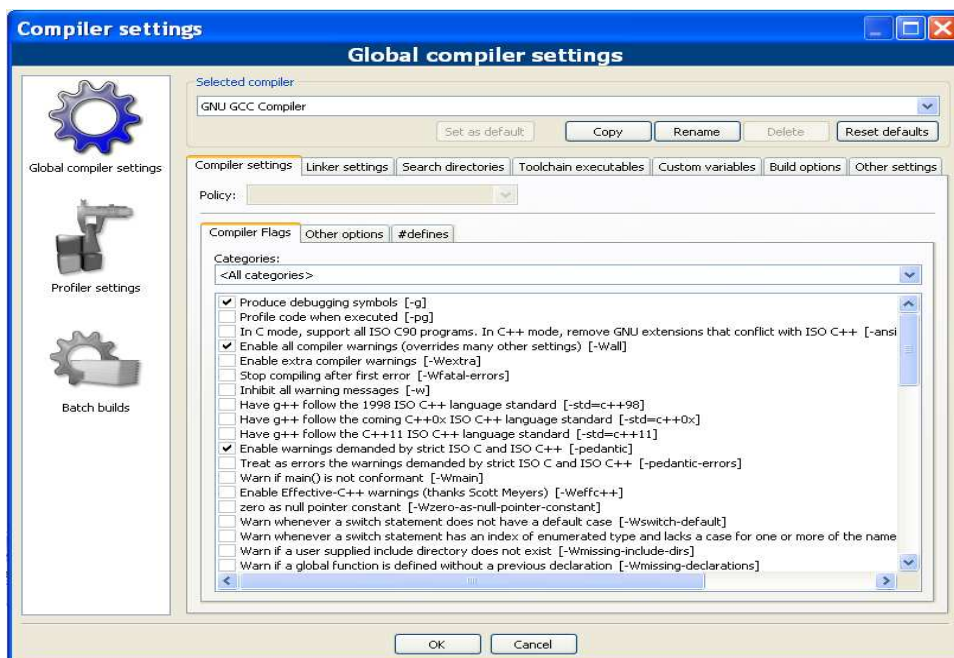
DEPURADOR (DEBUGGER)

Una de les possibilitats interessants que incorpora Code::Blocks, així com altres entorns de desenvolupament, és l'ús del depurador. Aquesta eina ofereix la possibilitat d'executar pas a pas el programa, examinar el valor de les variables, fer que el programa s'aturi quan arribi a una línia concreta (breakpoint), etc. Totes aquestes possibilitats resulten molt útils a l'hora de localitzar i corregir errors d'execució en els vostres programes.

Abans de poder fer servir el depurador en el vostre programa, si no ho heu fet abans, heu de configurar el compilador de la següent manera:

Configuració

–Entrem en el menú *Settings -> Compiler* i seleccionem la pestanya de *Compiler Settings*. Aquí li hem d'indicar que volem generar informació per a Debug activant l'opció “*Produce debugging symbols*”.



Penseu que el que fa el linkador és afegir informació de depuració al vostre programa executable, de tal manera que si el vostre programa no genera un executable (per exemple perquè té errors), no podreu depurar-lo. Aquesta informació incrementa considerablement la mida de l'executable final, per tant, un cop el programa està validat hem de tornar a compilar-lo sense l'opció d'afegir informació de debug.

Afegint Breakpoints

Un cop compilat per a debug el vostre programa, heu d'indicar on voleu que s'aturi l'execució afegint punts de ruptura o *breakpoints*. Ho podeu fer de les següents maneres:

- Fent click al costat esquerra de la línia on voleu que s'aturi. Quedarà marcat amb un punt vermell (podeu posar tants com cregueu convenient). Per treure els punts d'aturada només cal repetir la mateixa operació.
- Col·locant el cursor en la línia on voleu que s'aturi i després seleccionant al menú *Debug->Toggle breakpoint*

```

1  /* Programa que ens calcula el canvi a tornar */
2
3  #include <stdio.h>
4
5  int main() {
6      float preu, diners, canvi, ct;
7      int euros, centims, m2, m1, m50, m20, m5;
8      printf("Indica el preu del producte ");
9      scanf("%f", &preu);
10     printf("Indica els diners que dones ");
11     scanf("%f", &diners);
12     canvi=diners-preu;
13     // en aquest moment suposem que donem mes diners que el preu
14     // i per tant el canvi sera positiu
15     euros=(int)canvi; // agafem la part entera
16     ct=canvi-euros; // agafem els centims
17     centims=ct*100; // per a poder aplicar la divisio entera i el modul
18     // hem de treballar amb enters
  
```

Un cop posats aquests punts de ruptura, si executeu en mode debug (Menú *Debug->Start* o amb la tecla F8), veureu que l'execució s'atura en el primer punt que heu posat (abans d'executar la línia corresponent).

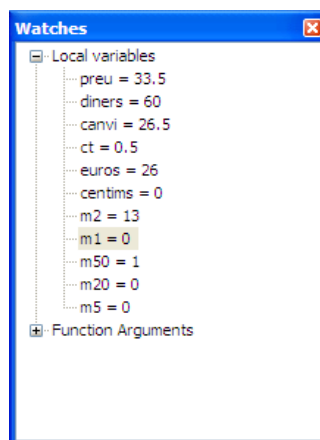
A partir d'aquí podeu avançar línia a línia amb diferents opcions:

- *Debug->Next line* (F7): executa cada línia pas a pas sense entrar als programes que es cridin.
- *Debug->Continue* (Ctrl+F7): executa el programa fins al següent breackpoint.
- *Debug->Step Into* (Shift+F7): executa cada línia però entrat dins dels programes i executant el seu codi corresponent línia a línia.
- *Debug->Step out* (Shift+Ctrl+F7): continua l'execució fins que finalitzi l'estructura de control actual (per exemple, si som dins d'un altre programa).

Inspeccionant les variables

Per comprovar la correcta execució del programa, quan estem aturats en una línia podem inspeccionar el valor que contenen les variables que estan visibles en l'àmbit actual.

Des del menú *Debug-> Debugging Windows*, podeu activar la finestra “*Watches*” que obrirà un nova finestra. Sobre aquesta a la part esquerra caldrà que indiqueu un a un els noms de les variables declarades al codi de les que voleu consultar el seu valor. Una altre opció és seleccionar el nom de la variable amb el cursor i amb el botó dret del ratolí triar la opció “*watch*”.



Per aturar el depurat trieu la opció del menú *Debug-> Stop Debugger*