



Victor 易記學 程式設計教學

[首頁](#)[最新消息](#)[常見問題](#)[好站連結](#)[酷學園討論區](#)[聯絡我們](#)

程式設計教學

[Python教學](#)[C語言教學](#)[C語言教學](#)[Visual C++ Express教學](#)[Code::Blocks 教學](#)[Dev C++ 教學](#)

相關網站

[Victor's 程式設計遇上小提琴
個人部落格](#)[Walele創意投票系統](#)

廣告

[廣告贊助商](#)[首頁](#) ▶ [C語言教學](#) ▶ [Code::Blocks 教學](#) ▶ Code::Blocks debugger教學

Code::Blocks debugger教學

作者是 Victor

週三, 11 二月 2009 23:54

Code::Blocks的Debugger

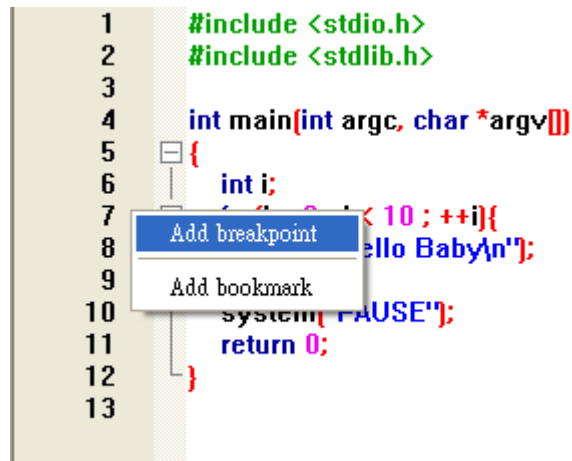
身為優秀的IDE，Code::Blocks自然也有提供Debugger的功能，我們以下介紹如下

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    int i;
    for(i = 0 ; i < 10 ; ++i){
        printf("Hello Baby\n");
    }
    system("PAUSE");
    return 0;
}
```

設定中斷點

設定中斷點的方式和大部份IDE差不多，如下圖所示，點擊行數數字旁的空白



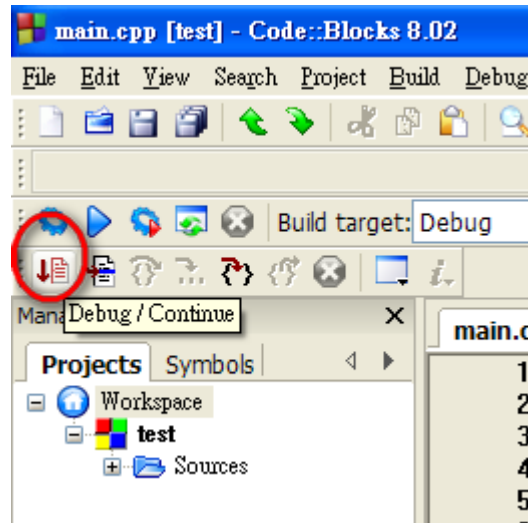
如果正確新增一個中斷點，會在該行數字旁邊看見一個小紅點

開始除錯

當新增完中斷點以後就可以開始除錯，在除錯之前請先注意，請注意你的Build Mode，請改回成Debug，或是編譯的參數裡要有 -g，要設定編譯參數，由選單Build->Options->Compiler->Code generation->Produce debugging symbols [-g]，打勾以後應該就可以進行

開春最強
動作鉅獻

除錯可以由選單的 Debug -> Start，或是按F8開始進行除錯



如果沒有錯的話，應該可以看到程式執行，並且有個黃色的箭頭停在小紅點

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(int argc, char *argv[])
5  {
6      int i;
7      for(i = 0; i < 10; ++i){
8          printf("Hello Baby\n");
9      }
10     system("PAUSE");
11     return 0;
12 }
13

```

除錯工具列

接下來我們介紹如何進行步進和除錯，以下是除錯用的工具列

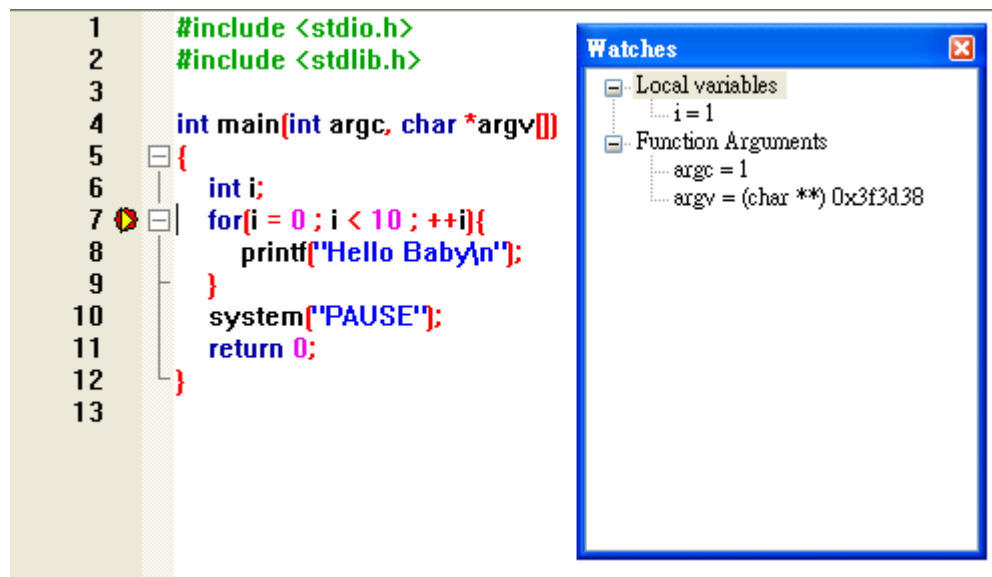


在此介紹每個按鈕的功能

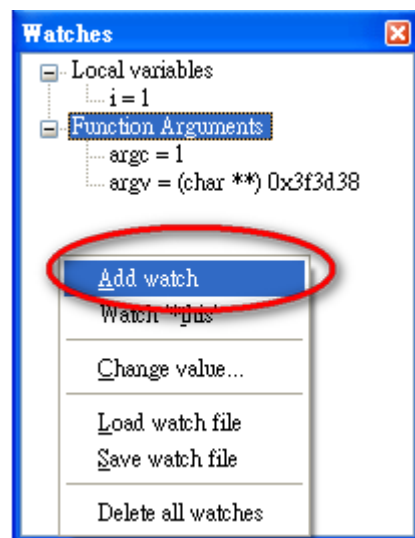
1. 開始除錯，也就是我們之前開始除錯所執行的功能
2. 執行程式，並且在游標位置中斷，當你不設中斷點，可是卻又想在再使用此功能
3. 下一行，如它圖示所暗示的，它並不會進入呼叫，而是直接跨過去，
4. 下一個指令，這是當你想看組合語言的下一個指令時所用的功能
5. 進入，此功能會跳入函數呼叫，如果你對函數裡的程式有興趣，就使
6. 離開，當你想跳出此函數呼叫到上一層時使用此功能
7. 中止除錯，如果你找到錯誤了，或是不想除錯了，就可使用此功能，
8. 除錯相關視窗，當你想查看CPU的暫存器內容，或是查看某個變數值
9. 資訊視窗，開啟一些比較瑣碎的資訊視窗

觀察變數

想要觀察變數的值，可以按右邊數過來第二個視窗圖案的按鈕，然後選Watc



想要增加觀察某個變數，可以按右鍵，Add watch



其它功能

其它功能，例如觀察呼叫堆疊、CPU暫存器等等，都在開啟觀察變數視窗的同樣按鈕中，在此就不多加贅述

[< 前一個](#)