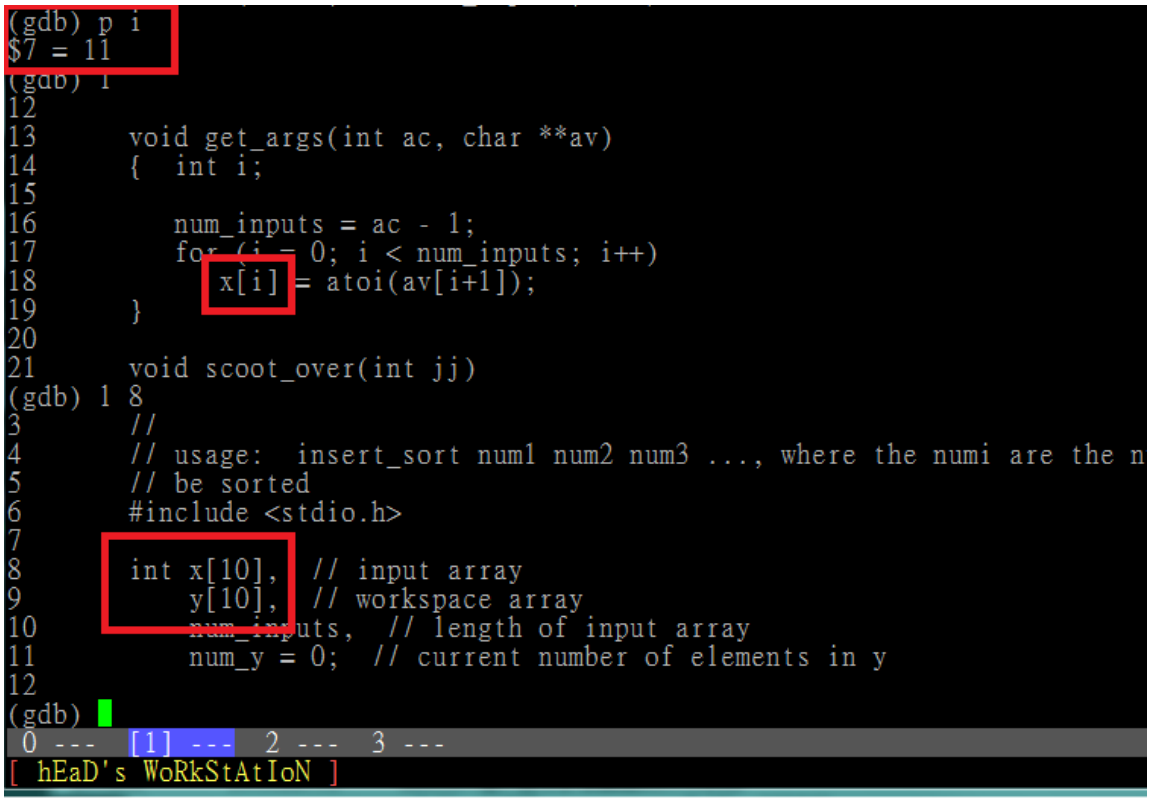
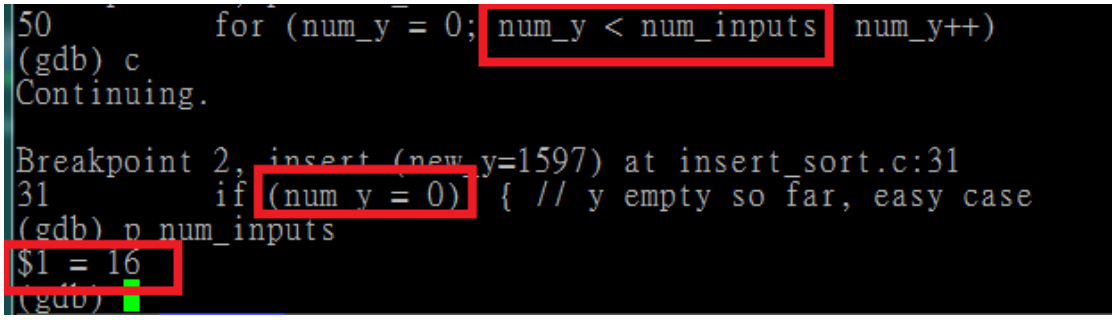
實驗一　Linux environment & debugging tool

9617145 許晏峻 資工 4C

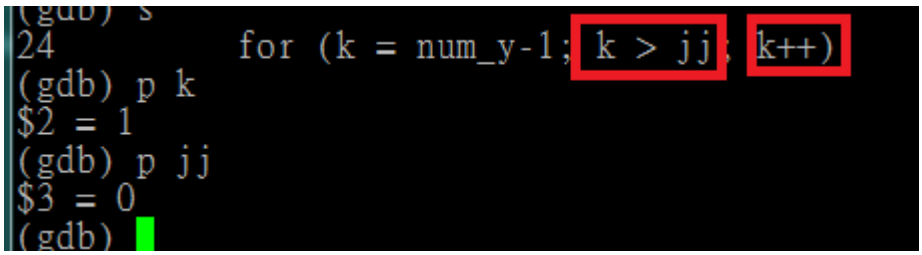
Project 1:



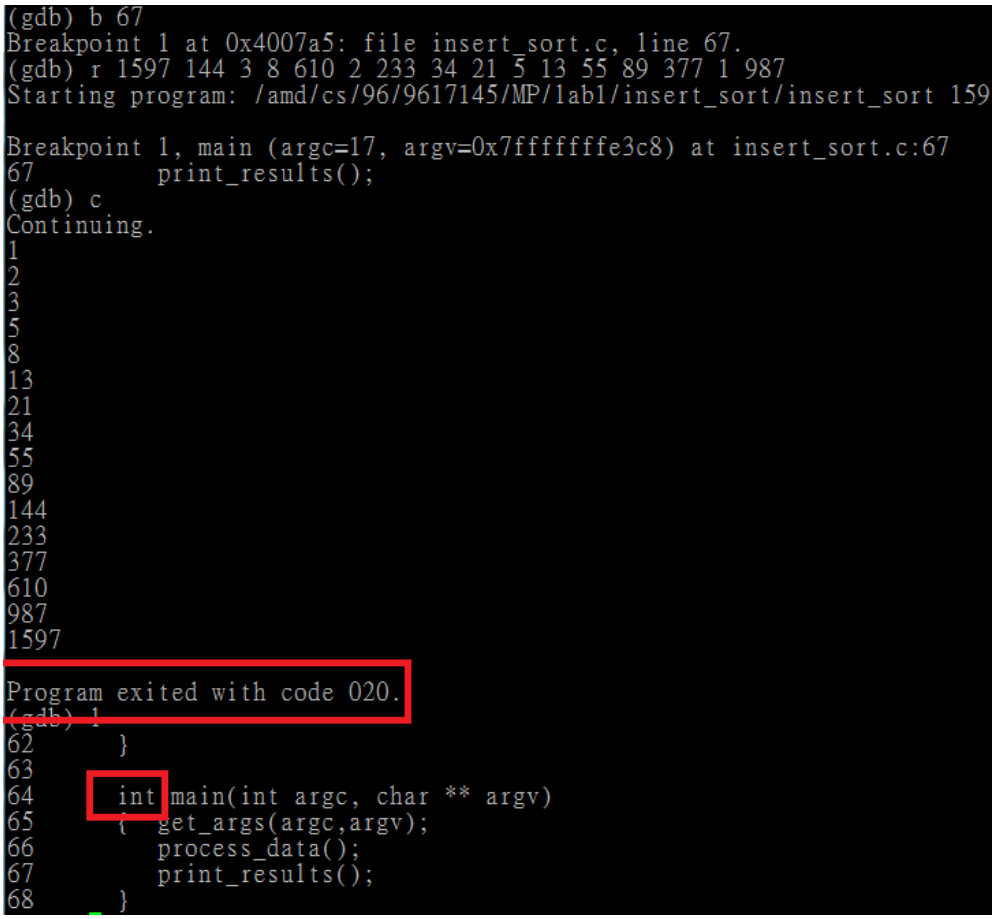
上圖：當 i=11 時，第 18 行 assign x[i]值，但在第 8 行宣告 x 空間只有 10，因此會 stack overflow



上圖：迴圈判斷 num_y < num_inputs ($1=16)，但迴圈中每次都會 if 判斷 num_y=0，少了一個 = ，變成 assign num_y=0，因此會無窮迴圈

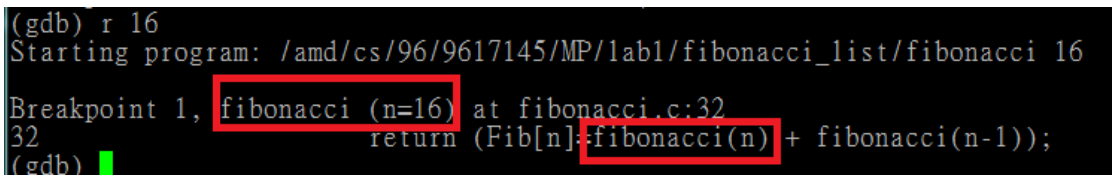


上圖：迴圈判斷 k>jj，但每次 k+1，會陷入無窮迴圈，應改成 k--

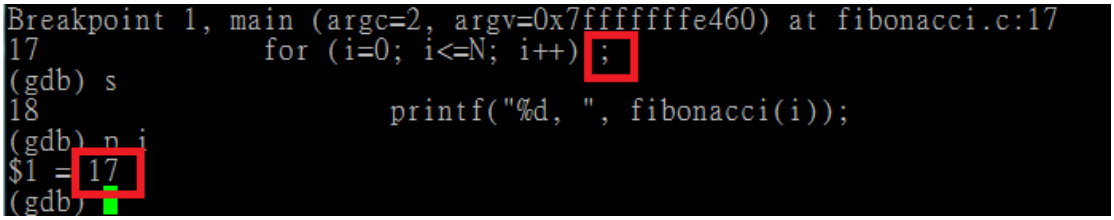


上圖：中斷點設在最後發現還有錯誤，因為 int main()沒有回傳一個 int，加上 return 0; 即可

Project 2:

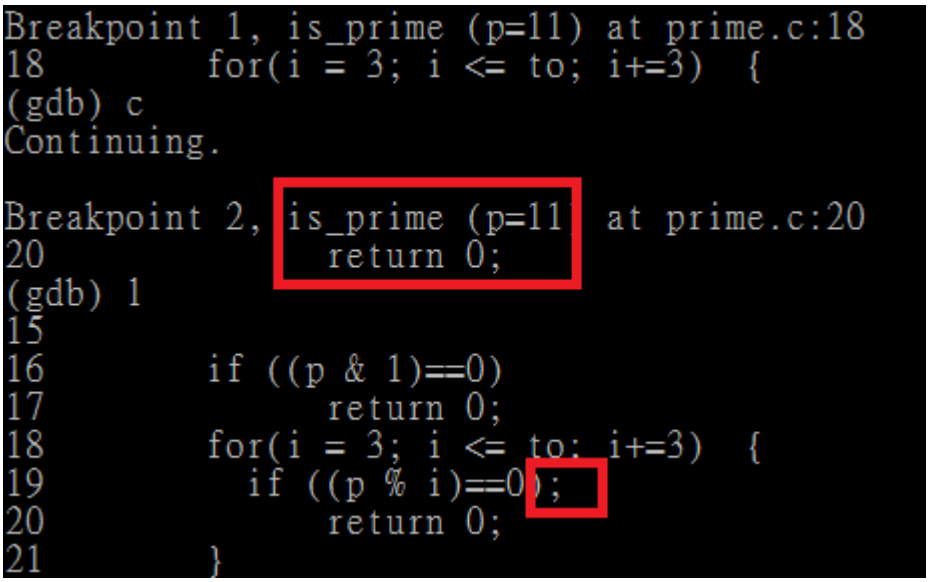


上圖：fibonacci(n=16)中又 call 一次 fibonacci(n=16)，很明顯會造成無限遞迴

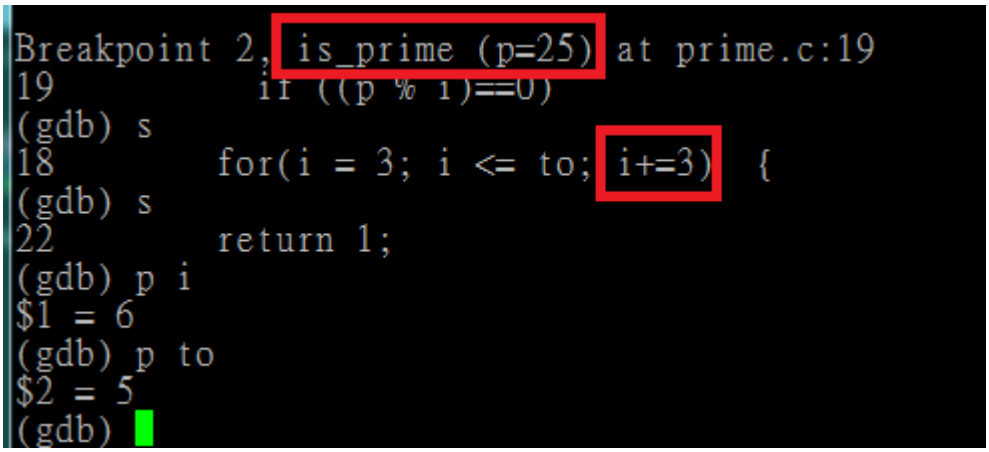


上圖：for(...)後多加了一個;，導致下一行的 printf(...)變成迴圈執行完才執行到，此時 i=17，已超過預計應該印出的陣列位置 Fib[0]~Fib[16]

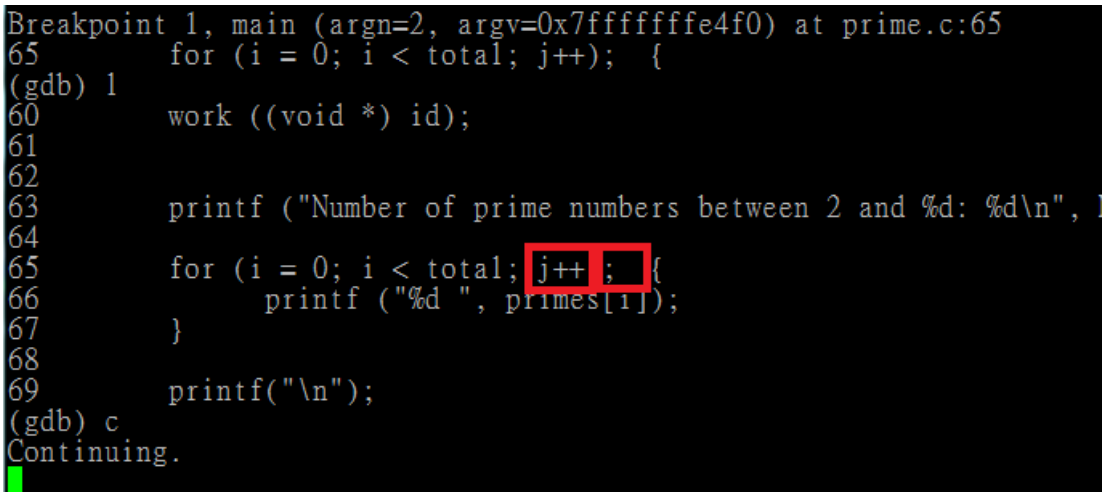
Project 3:



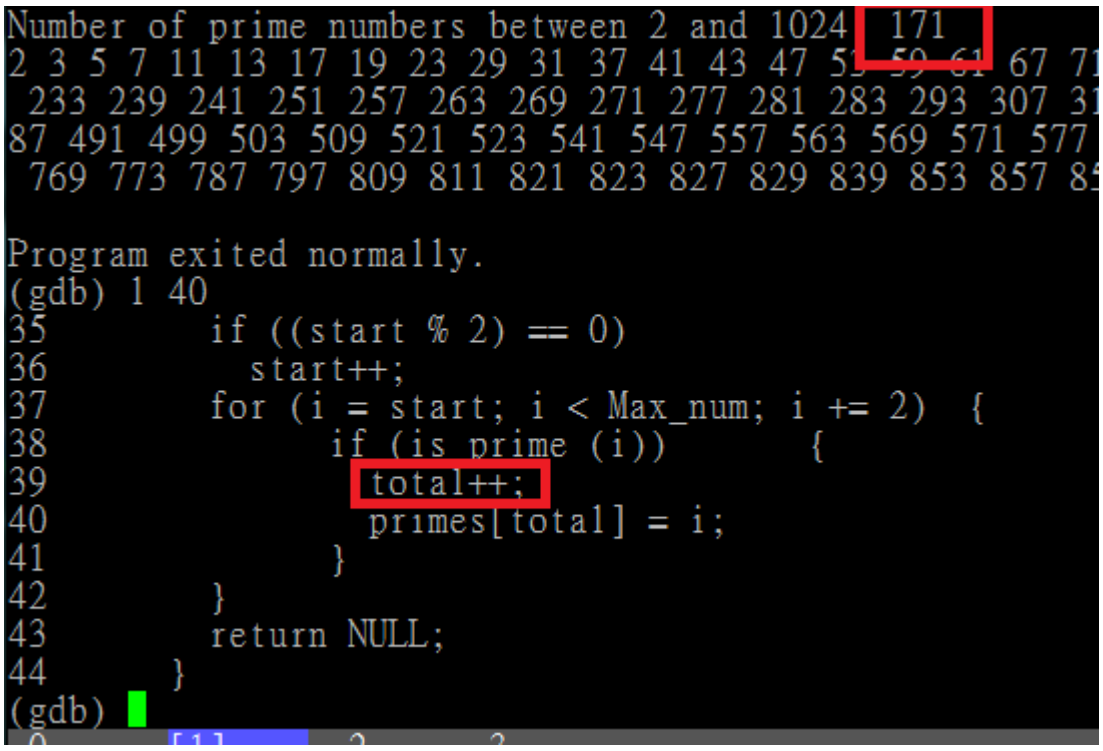
上圖：11 是質數，但卻 return 0，是錯誤的，因為第 19 行的 if 多加了一個;，導致 return 0 一定會被執行到



上圖：25 不是質數，但卻 return 1，是錯誤的，因為第 18 行的地方 i+=3，應該是做 i+=2 判斷每個奇數做(p%i)才對



上圖：在第 65 行後就進入無窮迴圈，因為迴圈判斷式用 i<total，但每次的 control 是 j++，應改成 i++。此外，for(...)後也多加一個; 因此 printf(...)會再迴圈結束後才印出，應該拿掉



上圖：程式已經可以正常執行，答案也正確，但是卻少印一個質數，因為第 39 行的 total++，少算第一個質數 primes[0]=2，或著也可以說 total 代表的是最後一個 prime 的陣列位置，因此質數數量應是 total+1，所以在迴圈結束後可再加一個++total