Use dump analysis to debug

1. Screenshots

(1) 例外視窗
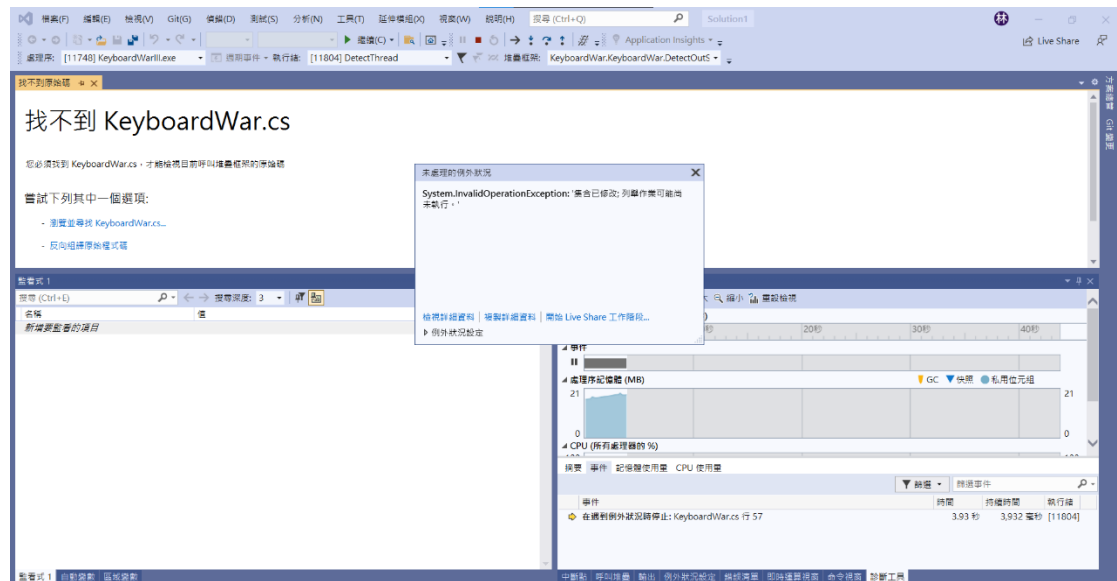
(2)　呼叫堆疊

▼ Main Thread

呼叫堆疊	▼ ⊅ ×
名稱	語言
[外部程式碼]	
⇨ KeyboardWarIII.exe!KeyboardWar.Program.Main() 第 20 行	C#

中斷點　呼叫堆疊　輸出　例外狀況設定　錯誤清單　即時運算視窗　命令視窗

▼ CreateMonsterThread

呼叫堆疊	▼ ⊅ ×
名稱	語言
[外部程式碼]	
⇨ KeyboardWarIII.exe!KeyboardWar.KeyboardWarForm.CreateMonsterThread.AnonymousMethod__6_0() 第 47 行	C#
[外部程式碼]	

中斷點　呼叫堆疊　輸出　例外狀況設定　錯誤清單　即時運算視窗　命令視窗

▼ DetectThread

呼叫堆疊	▼ ⊅ ×
名稱	語言
[外部程式碼]	
⇨ KeyboardWarIII.exe!KeyboardWar.KeyboardWar.DetectOutScreen() 第 57 行	C#
KeyboardWarIII.exe!KeyboardWar.KeyboardWarForm.CreateDetectThread.AnonymousMethod__7_0() 第 60 行	C#
[外部程式碼]	

中斷點　呼叫堆疊　輸出　例外狀況設定　錯誤清單　即時運算視窗　命令視窗

▼ UpdateThread

呼叫堆疊	▼ ⊅ ×
名稱	語言
[外部程式碼]	
⇨ KeyboardWarIII.exe!KeyboardWar.KeyboardWarForm.Update() 第 90 行	C#
[外部程式碼]	

中斷點　呼叫堆疊　輸出　例外狀況設定　錯誤清單　即時運算視窗　命令視窗

2. 推論出錯的地方在哪，還有推測原因及修正方法

　　我覺得有可能是 DetectThread 生成 child thread 之後，會不斷去偵測有沒有掉落到最下方的(death)的字母，有的話就會 call DetectOutScreen()，然後再呼叫 UpdateThread 從 KeyboardWar.Monster list 刪除 KeyboardWar.Monster 物件。但是因為同時有兩個以上的字母掉到螢幕外，所以同時呼叫UpdateThread去存取 KeyboardWar.Monster list，然後就造成 crash

(1)　　推測原因

先看到 crash 發生在 UpdateThred
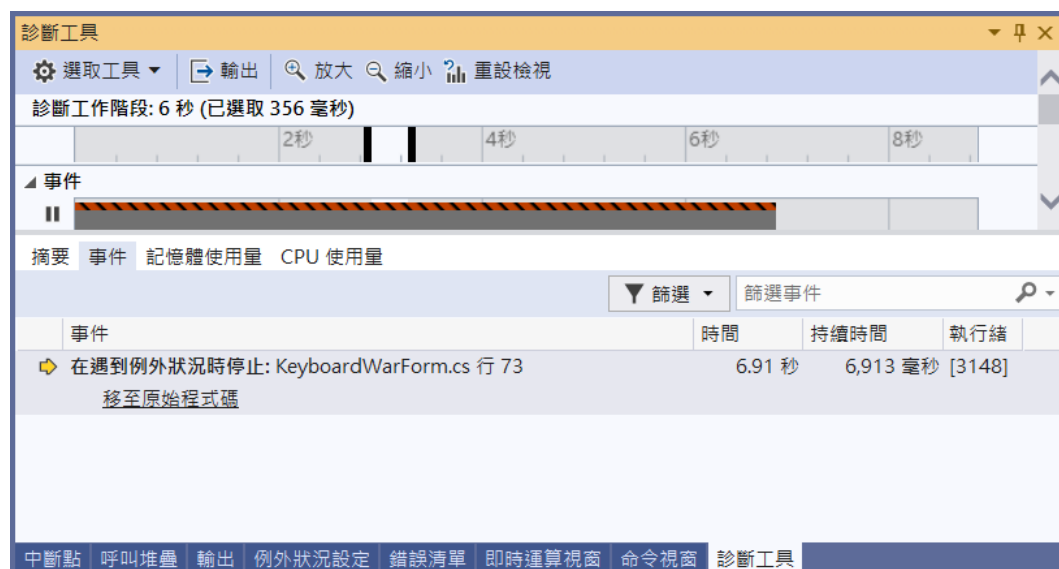
擷取記憶體快照

看一下 KeyboardWar.Monster list 的內容，可以看到最後兩個已經是 size = empty，但是還沒有從 KeyboardWar.Monster list 中移除，因此我想是不是有可能 UpdateThread 同時要移除這兩個物件，所以才造成 crash

(2)　修正方法

可能要再 Update()加上 lock，讓 Upate()只能讓一個 DetectThread 呼叫，就能一次刪除一個 KeyboardWar.Monster 物件，避免發生 race 的情況