

بسم الله الرحمن الرحيم
الفريق العربي للهندسة العكسية
تدريب عملي على إضافة دالة بطريقة الية

مقدمة:

في البداية هذا الدرس طبعا هو استكمال لدرس كتبة الأخ **Maxhak2000** عن إضافة دالة الي البرنامج بطريقة الية ولكن سوف اقوم في هذا الدرس بالتجريب العملي علي هذه الطريقة لانني عندما قمت بتجريبها لأول مرة وجدت ان هناك الكثير من الخطوات التي لم تذكر. وسوف نقوم بالتنفيذ علي برنامج الأخ **allko** في درسة التاسع في سلسلة تعلم الهندسة العكسية.

البداية:

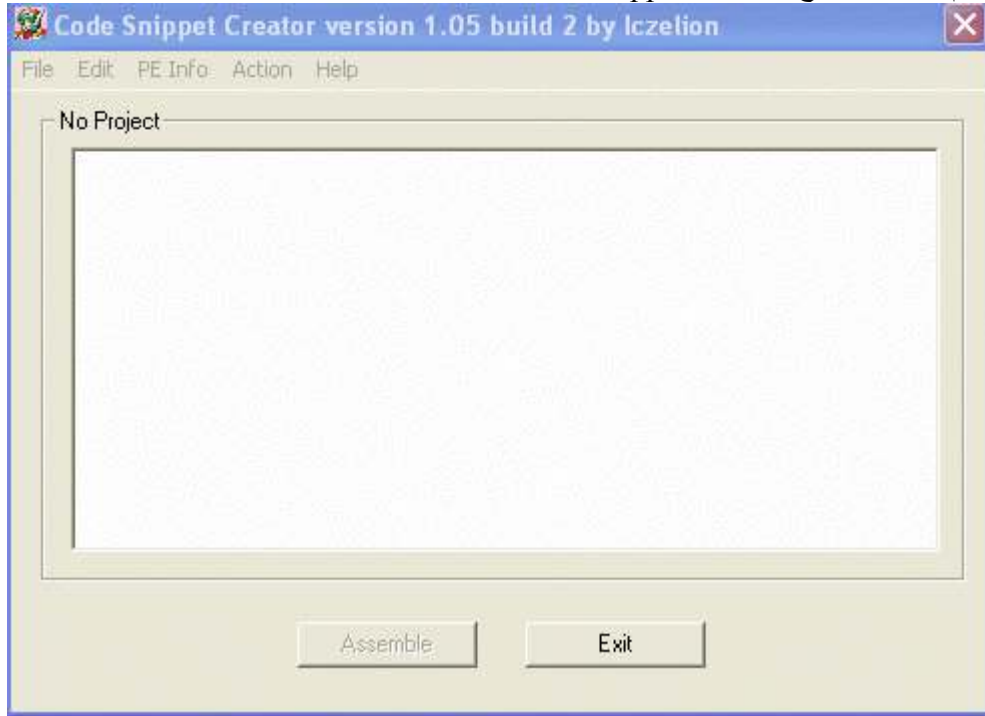
البرنامج المستخدم في إضافة الدوال بطريقة ألية يسمى Code Snippet Creator 1.05 Build 2 وصلة تحميل البرنامج :

<http://win32assembly.online.fr/files.sc.zip>

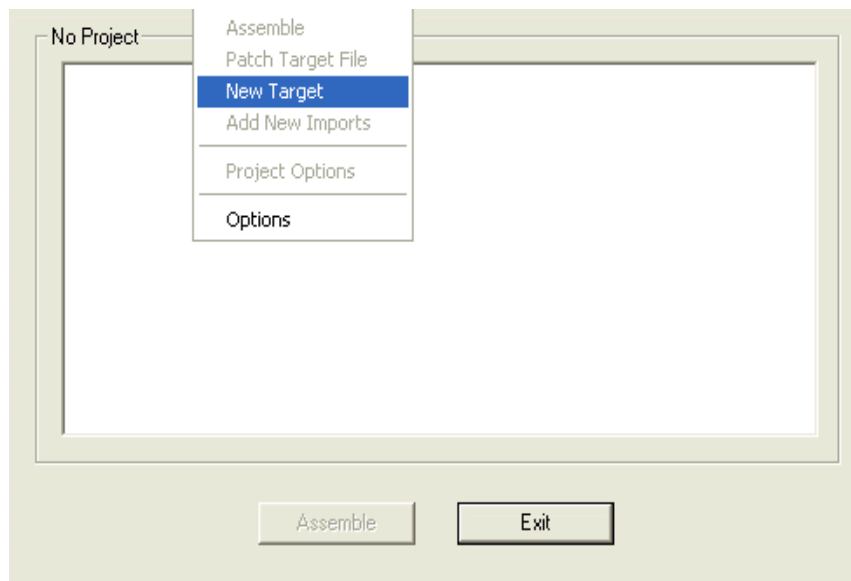
الخطوة الأولى:

طبعا هذه الطريقة للبرامج التي لا تحتوي علي مكتبة user32.dll أصلا . لذلك عند فحص البرنامج والمسمى monkey keygen 2 سوف تجد انه لا يحتوي علي هذه المكتبة و المسئولة عن دالة ال MessageBoxA والتي تعرض الرسالة التي نريد ادخالها في البرنامج. طبعا لن اشرح كيفية فحص البرنامج لان كل الخطوات مشروحة بالتفصيل في درس الأخ allko .

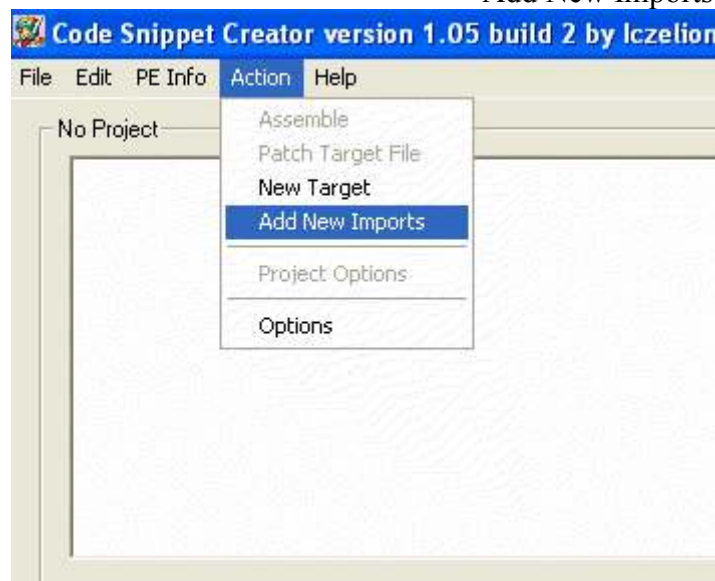
نقوم بتشغيل برنامج Code Snippet Creator



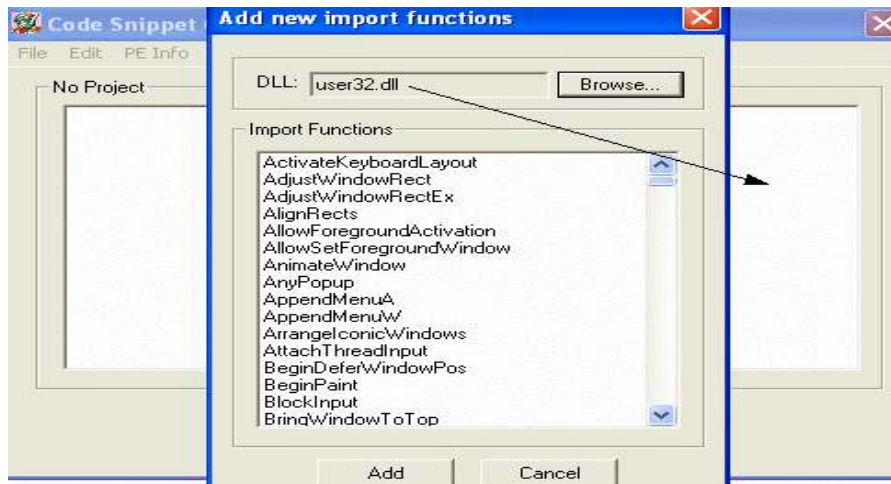
بعد ذلك نختار البرنامج الهدف من Action



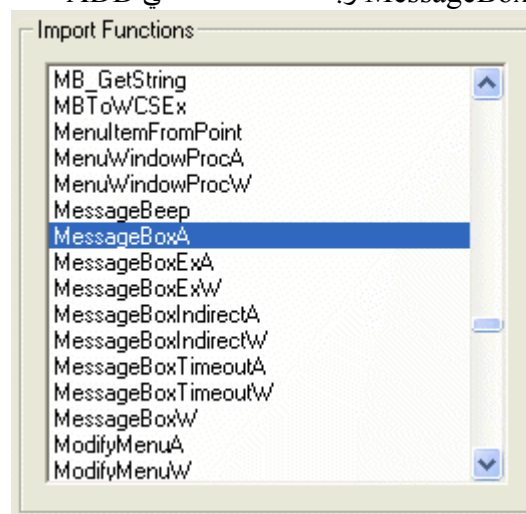
بعد ذلك نضغط على Add New Imports



نقوم بالبحث عن ملف ال user32.dll وسوف نجدة في مجلد الويندوز في ملف ال system32



سوف نقوم بأختيار دالة ال MessageBoxA وبعد ذلك اضغط على ADD



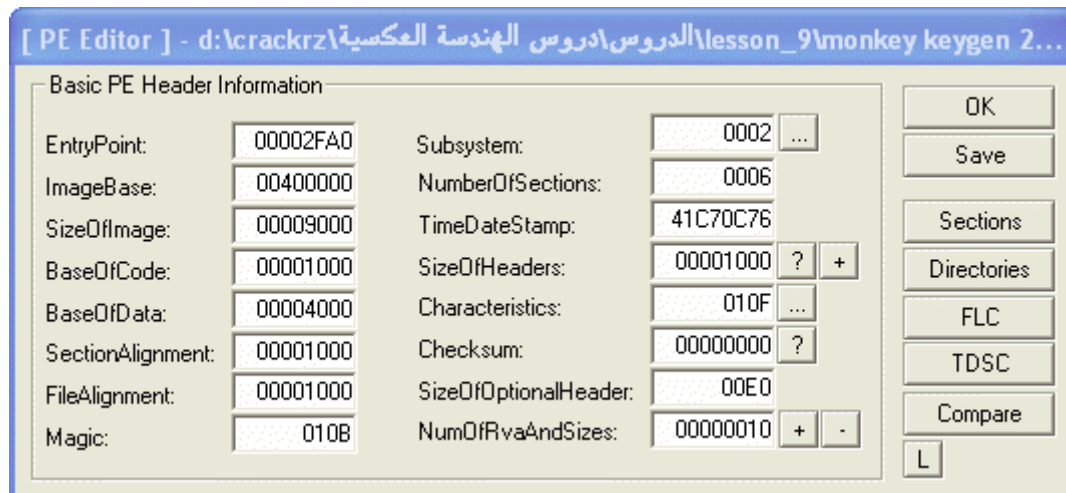
وبذلك انتهينا من الخطوة الأولى في هذه العملية.

الخطوة الثانية:

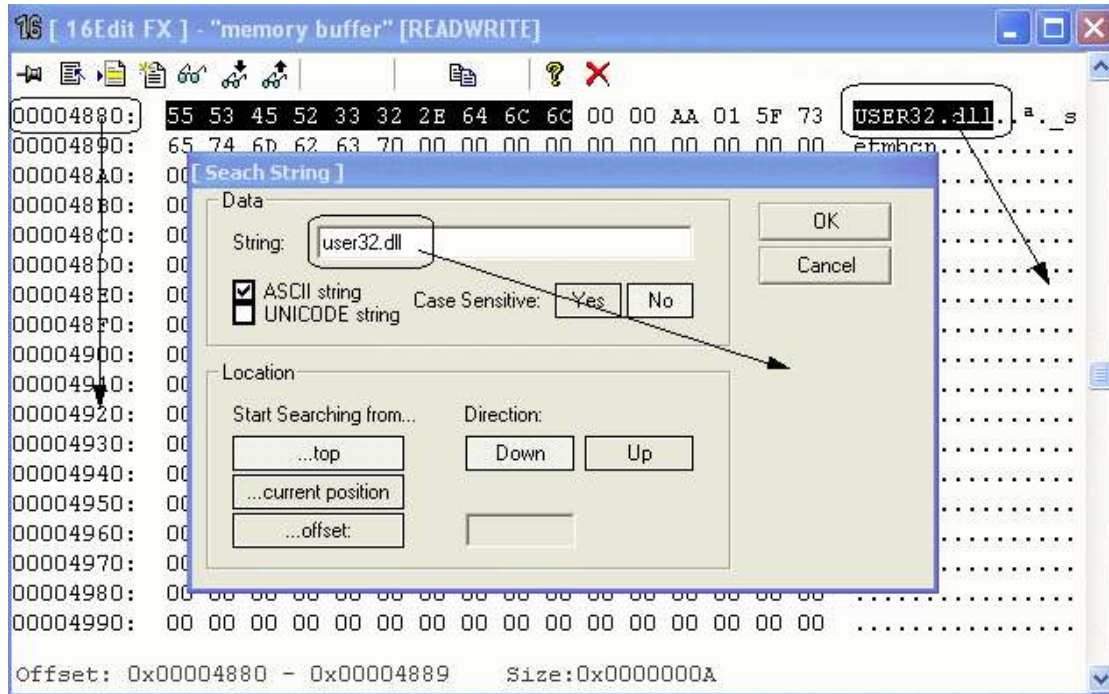
لاضافة MessageBoxA الي أي برنامج سوف نقوم بالخطوات التالية :

- 1- عنوان ال strings ل USER32.dll وذلك عن طريق برنامج ال LordPE
- 2- يلزمنا 3 strings الاولى نكتب فيها MessageBoxA والثانية تكون لعنوان هذه الرسالة و الثالثة نكتب فيها نص هذه الرسالة.
- 3- البحث عن مكان فارغ في كود البرنامج لنقوم بعمل code injection ثم نعيد السيطرة للبرنامج مرة أخرى.

*** لايجاد عنوان ال USER32.dll نقوم بفتح البرنامج الهدف في برنامج LordPE**

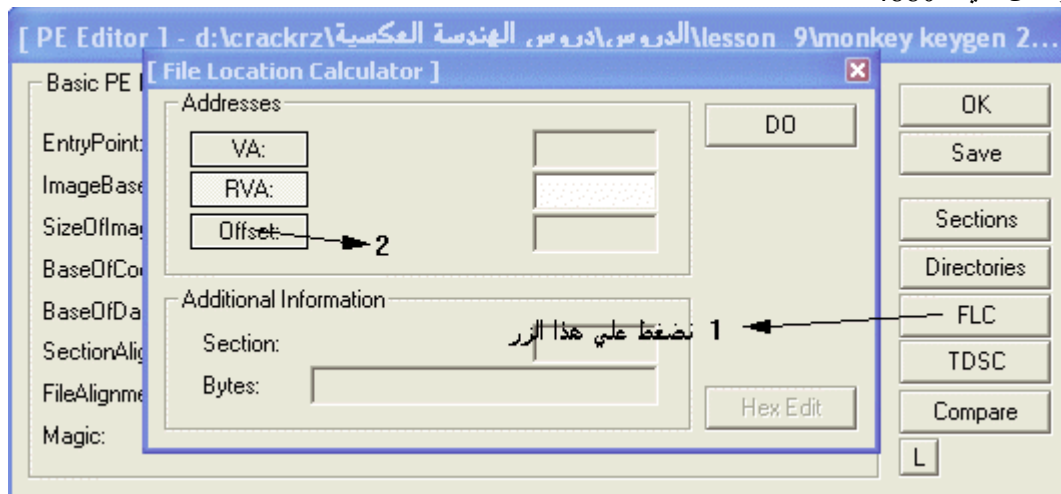


نقوم بأختيار Directories ثم بجانب ال ImportTable اضغط علي H لفتح محرر الهيكل.



عند البحث عن USER32.dll سوف نحدة عند ال Offset رقم 4880 بعد ذلك نقوم بحساب ال RVA وذلك باستخدام أيضا برنامج ال LordPE

سوف نضغط علي FLC حتي تخرج لنا نافذة تعتبر الة حاسبة لبرنامج LordPE ثم نضغط علي offset وندخل القيمة 4880



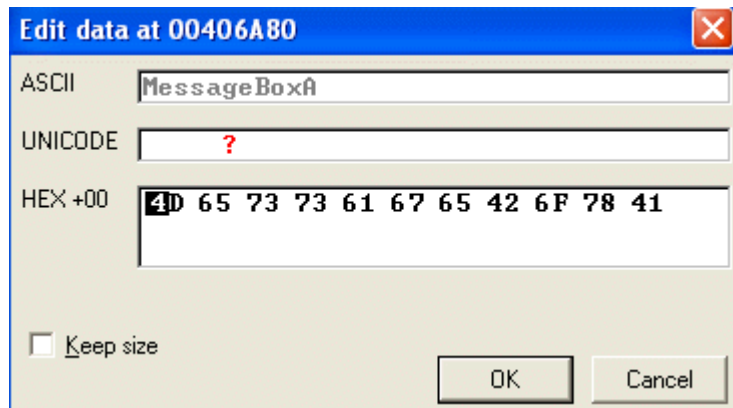
ثم نضغط علي DO سنجد ان قيمة ال RVA تساوي 4880 وهذه حالة شاذة ان يكون قيمة ال RVA هي نفسها قيمة ال Offset اي ليس من المعتاد تساوي القيمتين.

* كتابة ال strings الخاصة بال MessageBoxA :

سوف نقوم بتشغيل البرنامج في ال Ollydbg وللتأكد من اضافة الدالة MessageBoxA اضغط علي ctrl+N سوف تجد التالي

00404184	.rdata	Import	MSVCRT._acmdln	
00404194	.rdata	Import	MSVCRT._adjust_fdiv	
00408017	iImport	Import	user32.AllowForegroundActivation	
00408064	iImport	Import	user32.AllowForegroundActivation	
004041A8	.rdata	Import	MSVCRT._controlfp	
004041B4	.rdata	Import	MSVCRT._CxxFrameHandler	
004041B0	.rdata	Import	MSVCRT._dillonexit	
004041C4	.rdata	Import	USER32.DrawIcon	
004041D4	.rdata	Import	USER32.EnableWindow	
004041A4	.rdata	Import	MSVCRT._except_handler3	
00404180	.rdata	Import	MSVCRT.exit	
004041BC	.rdata	Import	MSVCRT._exit	
004041C8	.rdata	Import	USER32.GetClientRect	
00404188	.rdata	Import	MSVCRT._getmainargs	
00404000	.rdata	Import	KERNEL32.GetModuleHandleA	
00404004	.rdata	Import	KERNEL32.GetStartupInfoA	
004041CC	.rdata	Import	USER32.GetSystemMetrics	
0040418C	.rdata	Import	MSVCRT._inittterm	
004041D0	.rdata	Import	USER32.IsIconic	
004041D8	.rdata	Import	USER32.LoadIconA	
0040801B	iImport	Import	user32.MessageBoxA	
00408068	iImport	Import	user32.MessageBoxA	
00402FA0	.text	Export	<ModuleEntryPoint>	
004041AC	.rdata	Import	MSVCRT._onexit	
00404198	.rdata	Import	MSVCRT._p_commode	
0040419C	.rdata	Import	MSVCRT._p_fmde	
004041DC	.rdata	Import	USER32.SendMessageA	
004041B8	.rdata	Import	MSVCRT._setmbcp	
00404190	.rdata	Import	MSVCRT._setusermatherr	
004041A0	.rdata	Import	MSVCRT._set_app_type	
0040417C	.rdata	Import	MSVCRT._XcptFilter	

نذهب الي قسم ال Binary ونختار اي عناوين فارغة مثلا العنوان 0406A80 ونقوم بكتابة MessageBoxA



ونختار عناوين اخري لعنوان الرسالة و نص الرسالة مثل 0406A60 و 0406A80 ونكتب مثلا عنوان الرسالة AT4RE ونص الرسالة Arab Team For Reverse Engineering

Address	Hex dump	ASCII
00406A30	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406A40	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406A50	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406A60	41 54 34 52 45 00 00 00 00 00 00 00 00 00 00 00	AT4RE.....
00406A70	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406A80	41 72 61 62 20 54 65 61 6D 20 46 6F 72 20 52 65	Arab Team For Re
00406A90	76 65 72 73 65 20 45 6E 67 69 6E 65 65 72 69 6E	verse Engineerin
00406AA0	67 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	g.....
00406AB0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406AC0	00 00 4D 65 73 73 61 67 65 42 6F 78 41 00 00 00	..MessageBoxA...
00406AD0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406AE0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406AF0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406B00	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406B10	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
00406B20	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	

* البحث عن مكان فارغ في البرنامج لكتابة الكود:

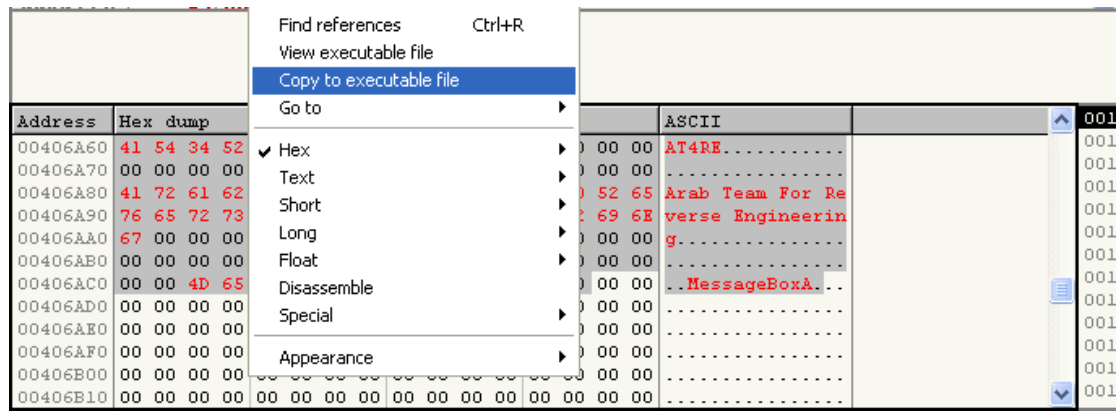
نقوم بالبحث عن اي مكان فارغ في البرنامج لنقوم بكتابة الكود المسؤول عن اظهار الرسالة مثلا اختار العنوان 004031FF واكتب الكود التالي :

004031FD	00 DB 00	
004031FE	00 DB 00	
004031FF	60 PUSHAD	
00403200	68 PUSH monkey_k.00404880	ASCII "USER32.dll"
00403205	E8 CALL kernel32.LoadLibra	
0040320A	68 PUSH monkey_k.00406AC2	ASCII "MessageBoxA"
0040320F	50 PUSH EAX	
00403210	E8 CALL kernel32.GetProcAd	
00403215	6A PUSH 0	
00403217	68 PUSH monkey_k.00406A60	ASCII "AT4RE"
0040321C	68 PUSH monkey_k.00406A80	ASCII "Arab Team For Reverse Engineering"
00403221	6A PUSH 0	
00403223	FF CALL EAX	
00403225	61 POPAD	
00403226	56 PUSH ESI	
00403227	00 ADD [BYTE DS:EAX],AL	
00403229	00 ADD [BYTE DS:EAX],AL	
0040322B	00 ADD [BYTE DS:EAX],AL	
0040322D	00 DB 00	
0040322F	00 DB 00	

بعد ذلك نريد ان نبحث عن مكان في البرنامج لنغير منه مجري البرنامج الي مكان الكود المدخل ثم بعد ذلك نعيد السيطرة للبرنامج مرة اخري.

سوف نختار العنوان 402FC1 عند الأمر Push ESI وذلك لكي تظهر الرسالة في بداية تشغيل البرنامج ونحول هذا الأمر الي jmp 004031FF الي بداية الكود وفي نهاية الكود نضع قفزة للعودة للعودة jmp 00402FCB

نقوم الآن بحفظ البرنامج الجديد بأسم مختلف وبعد ذلك نحفظ تغييرات قسم ال Binary ثانيا



وبعد ذلك نقوم بتشغيل البرنامج الجديد وهذه هي النتيجة



النهاية :

طبعاً هذا الدرس لم يضيف جديد علي دروس الأخ *allko* ودرس الأخ *Maxhak2000* لكن حاولت ان ارتب الأفكار و الخطوات التي قاموا بها وان اربط بين الطريقتين.

وفي النهاية شكر خاص لجميع أعضاء الفريق العربي للهندسة العكسية.
أخوكم *Ahmed18/AT4RE*