JUAN EL VIEJO
CRACKSLATINOS

Programa	Ejemplo
Herramientas	IDA-wxDev-C++
Protección	
Objetivo	Reversear

Ejecutamos Ejemplo.exe, para ver qué hace el programa: a+b y a>b
Vemos que nos pide ingresar dos números, saca el valor de la suma y comprueba:
Ya lo veremos en IDA. Para que se vea un poco más claro intentaremos que el número ingresado quede: Ingrese el valor de a: numero

```
Ingrese el valor de a:
4
Ingrese el valor de b:
3
El valor de la suma es 7:

El valor de a es mayor que el de b
```

Lo volvemos a ejecutar: a+b y b>a

```
Ingrese el valor de a:
3
Ingrese el valor de b:
4
El valor de la suma es 7:

El valor de b es mayor que el de a
```

Lo volvemos a ejecutar: a+a y a=b

```
Ingrese el valor de a:
4
Ingrese el valor de b:
4
El valor de la suma es 8:

Son iguales
```

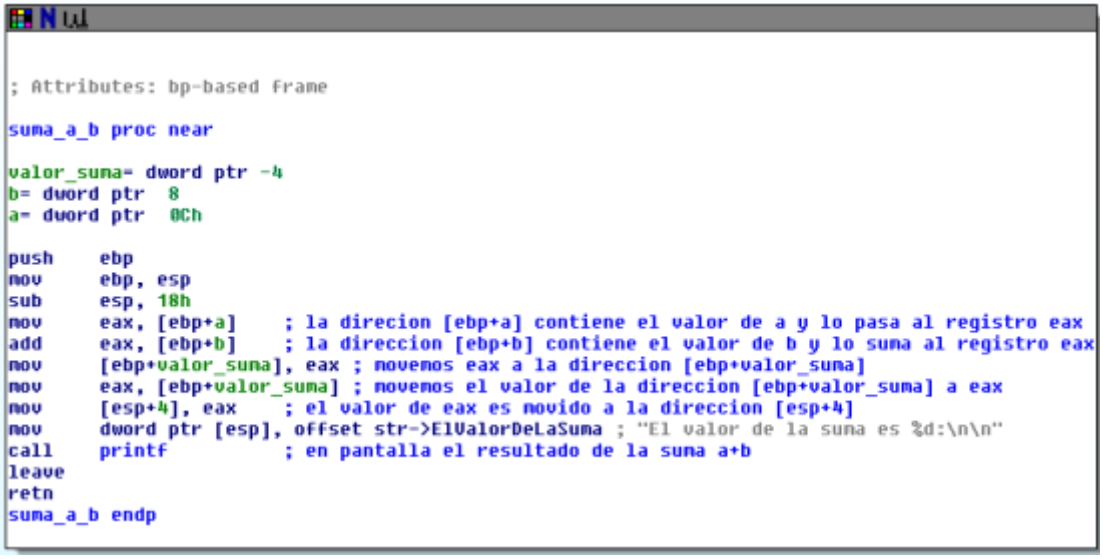
Información:
Abrimos el programa en IDA.
Le damos a: W y nos lo acopla a la pantalla (para volver atrás le damos a 1) y volvemos a tenerlo como al principio.
Para no tener que mirar todo el código y ver donde hace la suma y las comparaciones, nos vamos a la derecha, según la pantalla.
Vemos la parte que nos interesa de dos maneras: como lo da IDA y con el cambio (aquí he puesto la imagen del cambio; pero es sacada al final).

Fin Información:

Aquí se ve todo lo que he deducido antes. Con unas modificaciones para su buen entendimiento. Maestro Ricardo: si algunos de esos comentarios no tienen concordancia, rectifícame, gracias.

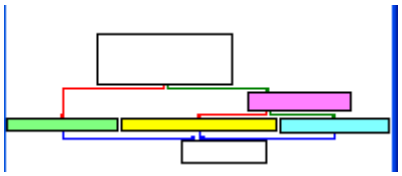
```
; Attributes: bp-based frame

; int __cdecl main(int argc, const char **argv, const char **envp)
_main proc near

var_C= dword ptr -0Ch
numero_b= dword ptr -8
numero_a= dword ptr -4
argc= dword ptr  8
argv= dword ptr  0Ch
envp= dword ptr  10h

push    ebp
mov     ebp, esp
sub     esp, 18h
and     esp, 0FFFFFFF0h
mov     eax, 0
add     eax, 0Fh
add     eax, 0Fh
shr     eax, 4
shl     eax, 4
mov     [ebp+var_C], eax
mov     eax, [ebp+var_C]
call    ___chkstk
call    ___main
mov     dword ptr [esp], offset str->IngreseElValorD ; "Ingrese el valor de a:\n"
call    printf          ; informacion en pantalla
lea     eax, [ebp+numero_a] ; direccion para el numero a
mov     [esp+4], eax
mov     dword ptr [esp], offset str->D ; "%d"
call    scanf           ; entramos el numero a
mov     dword ptr [esp], offset str->IngreseElValo_0 ; "Ingrese el valor de b:\n"
call    printf          ; informacion en pantalla
lea     eax, [ebp+numero_b] ; direccion para el numero b
mov     [esp+4], eax
mov     dword ptr [esp], offset str->D ; "%d"
call    scanf           ; entramos el numero b
mov     eax, [ebp+numero_b]
mov     [esp+4], eax
mov     eax, [ebp+numero_a]
mov     [esp], eax
call    suma_a_b        ; dirccion donde calcula la suma
mov     eax, [ebp+numero_b]
mov     [esp+4], eax
mov     eax, [ebp+numero_a]
mov     [esp], eax
call    comparacion_a_b ; direccion donde hace las comparaciones
call    _getch
mov     eax, 0
leave
retn
_main endp
```

Doble clic en la primara, a esto le llamaremos: suma_a_b, la explicación en la imagen.

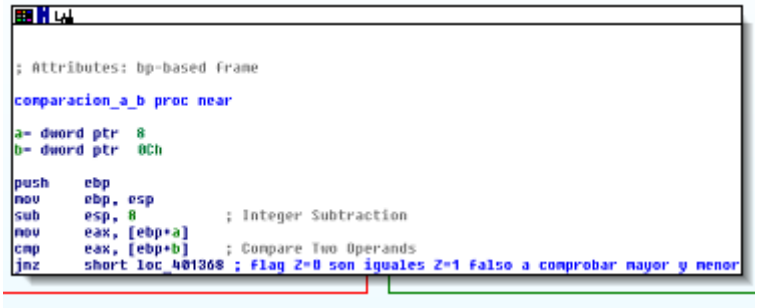


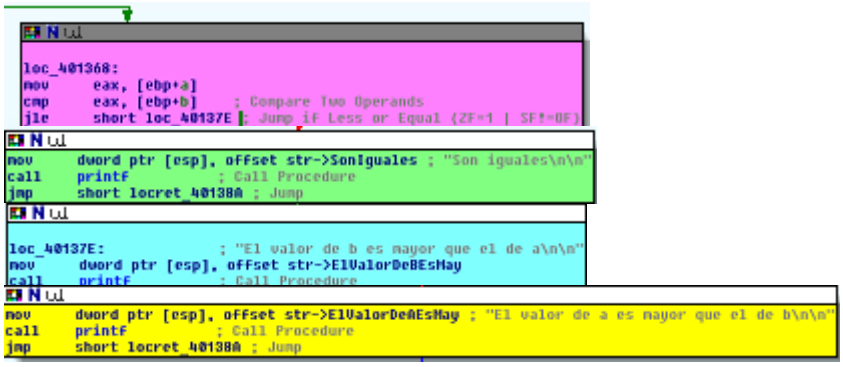
```
; Attributes: bp-based frame

suma_a_b proc near

valor_suma= dword ptr -4
b= dword ptr  8
a= dword ptr  0Ch

push    ebp
mov     ebp, esp
sub     esp, 18h
mov     eax, [ebp+a]    ; la direcion [ebp+a] contiene el valor de a y lo pasa al registro eax
add     eax, [ebp+b]    ; la direccion [ebp+b] contiene el valor de b y lo suma al registro eax
mov     [ebp+valor_suma], eax ; movemos eax a la direccion [ebp+valor_suma]
mov     eax, [ebp+valor_suma] ; movemos el valor de la direccion [ebp+valor_suma] a eax
mov     [esp+4], eax    ; el valor de eax es movido a la direccion [esp+4]
mov     dword ptr [esp], offset str->ElValorDeLaSuma ; "El valor de la suma es %d:\n\n"
call    printf          ; en pantalla el resultado de la suma a+b
leave
retn
suma_a_b endp
```

Doble clic en la segunda, una imagen pequeñita para ver lo que hace.



A esto lo llamamos: comparacion_a_b





Ya tenemos toda la información, sólo nos queda programar para reversarlo.
Sierro IDA.
Abro wxDev-C++, cuando me canse de este, instalo el tuyo que se sigue mejor.
Mando todo lo que he hecho en un rar.

Hasta aquí llegué, un saludo a toda la lista. Juan