REVERSE ENGINEERING

04 – C++ FOR REVERSE ENGINEERS

REVERSING C++ OBJECTS

Lab Description: Reversing software requires the ability to identify and effectively analyze a wide variety of code constructs and patterns. For this lab, the student is required to reverse engineer a C++ program that uses objects.

Lab Environment: IDA Pro Educational

Lab Files that are Needed: ReversingCPP.exe.

Analyze the provided lab file and answer the following questions:

1. How many objects are created?
2. What is the size of that object/what are the sizes of those objects?
3. Does the first class have a virtual function? Include a screenshot with answer.
4. Does the second class inherit the first class? Include a screenshot with answer.
5. What is Jerry's number (ID)?
6. What is Bruce's number (ID)?
7. Is Jerry a base object or a derived object? Include a screenshot with answer.
8. Is Bruce a base object or derived object? Include a screenshot with answer.

WHAT TO SUBMIT

Submit a Microsoft Word document or PDF that includes answers to the questions posed along with screenshots demonstrating the installation of the virtual machine.

1. There are 2 objects initialized in main.

2. The sizes of the objects are 32, and 72.

3. It does have a virtual function. You can tell by calls to registers.

```
mov     eax, [ebp+var_4]
mov     edx, [eax]
mov     ecx, [ebp+var_4]
mov     eax, [edx]
call    eax
```

4. The second class inherits from the first class.

```
mov     eax, [ebp+var_4]
mov     dword ptr [eax], offset ??_7derivedClass@@6B@ ; const derivedClass:
mov     eax, [ebp+var_4]
mov     esp, ebp
```

5. Jerry's id is 492734.

6. Bruce's id is 923543.

7. Jerry is the base object.

```
.text:00CF1186 call    sub_CF19F0      ; Call Procedure
.text:00CF1188 mov     edx, [ebp+var_10]
.text:00CF118E mov     eax, [edx]
.text:00CF1190 mov     ecx, [ebp+var_10]
.text:00CF1193 mov     edx, [eax]
.text:00CF1195 call    edx             ; Indirect Call Near Procedure
.text:00CF1197 push    0E1797h
```

```
RAX 0000000000D0E380 "" .rdata:const baseClass::`vftable'
RBX 00000000010C5000 "" TIB[0000081C]:010C5000
RCX 0000000001397430 "" debug029:01397430
```