Symbol / Name	Address	What is in the memory of the computer Byte 1	Byte 2	Bite3	Bite4	Value As Interpreted by Type	Comments
Code Block							
PrintAllStaticAndGlobalAddresses	008A1019	??	??	??	??	A bunch of binary commands	No Parameters
	…	??	??	??	??		
PrintAllFunctionAddresses	008A1131	??	??	??	??	A bunch of binary commands	No Parameters
	…	??	??	??	??		
Cat::Speak	008A12A8	??	??	??	??	A bunch of binary commands	Hidded Cat* this parameter
	…	??	??	??	??		
PassingParameters	008A12EE	??	??	??	??	A bunch of binary commands	See Declaration for parameters
	…	??	??	??	??		
Dog::Speak	008A1492	??	??	??	??	A bunch of binary commands	Hidded Dog* this parameter
	…	??	??	??	??		
main	008A14B0	??	??	??	??	A bunch of binary commands	No Parameters
	…	??	??	??	??		
Animal::Speak	008A1532	??	??	??	??	A bunch of binary commands	Hidded Animal* this parameter
	…	??	??	??	??		
AnotherFunc	008A1578	??	??	??	??	A bunch of binary commands	No Parameters
	…	??	??	??	??		
More including constructs but we can't get their address I don't believe.							
Static Global Block							
int Animal::AnimalCount	008B5384	00	00	00	00	Initially 0	
Cat Pablo / Pablo Virtual Table Ptr	008B5388	??	??	??	??	Address for Virtual Table	Size is 16 (4 for Virtual table ptr, 4 for int Age, 4 for int Lives, 4 for int Sleep hours)
->Age	008B538C	00	00	00	01	1	
->Lives	008B5390	00	00	00	9	9	
->SleepHours	008B5394	00	00	00	14	20	
Stack - Functions are laid out in a decending order with values acending within each function							
Stack - Main Function							
Local Variables							
Animal* myAnimalPtr	009BF714	??	??	??	??		
Dog Gizmo / Virtual Table Ptr	009BF720	??	??	??	??	Address for Virtual Table	
->Age	009BF724	00	00	00	01	1	
->NumberOfTreats	009BF728	00	00	00	00	0	
Stack - PassingParameters Function							
Parameters							
int MyIntParam	009BF5F0	00	00	00	01	1	

Animal* AnyAnimalParam	009BF5F4	00	9B	F7	20	009BF720	This is the address of Gizmo
Cat CatOneParam / Virtual Table Ptr	009BF5F8	??	??	??	??	Address for Virtual Table	
->Age	009BF5FC	00	00	00	01		
->Lives	009BF600	00	00	00	9		
->SleepHours	009BF604	00	00	00	14		
MyMiddleParam	009BF608	00	00	00	04		4
MyMiddleParam2	009BF60C	00	00	00	05		5
Cat& CatTwoParam	009BF610	00	8B	53	88	008B5388	This is the address of Pablo
MyLastParam	009BF614	00	00	00	03		3
Local Variables							
int LocalInt	009BF5D0	00	00	00	02		2
int* myPtr	009BF5C4	00	9B	F6	10	009BF610	This is the address of Cat& CatTwoParam that we calculated
Dog* myDogPtr	009BF5B8	00	D3	3C	10	00D33C10	The address of the dog created on the Heap
Stack - AnotherFunc Call 1							
int myNewInt	009BF4A8	00	00	00	05		
Stack - AnotherFunc Call 2							
int myNewInt	009BF604	00	00	00	05		This is the same address as CatOneParm->SleepHours because that memor location is being reused after being removed from the stack
Heap							
New Dog() / Virtual Table Ptr	00D33C10	??	??	??	??	Address for Virtual Table	
->Age	00D33C14	00	00	00	01		
->NumberOfTreats	00D33C18	00	00	00	00		