Sarah Dunbar
Theory Assignment 1

Part 1: There are three variables used in this program. Two of them are of native type int and one is received from the caller. The integer variables are called _X and _Y in my program. However, these variables are actually different fields that are apart of the IntPoint2D class, so all objects stored in memory will be of that type. To start this memory diagram there will be a memory space that holds a pointer to where the different fields of our object will be stored, and one variable that will store a value received from a caller, but before any function call, the values start unknown:

Address	Type	Name	Value	
0x00A	IntPoint2D	point_a	unknown	
0x00B	String[]	args	unknown	
0x00C				
0x00D				
0x00E				
0x00F				
0x00G				
0x00H				
0x00I				

Part 2: Because IntPoint2D is not a native type in java, the value that this memory location will store is actually a pointer to an allocated space (that is the length of the object) which will store the different fields of our non-native object. The memory addresses highlighted in blue are the ones allocated for the values of the IntPoint2D objects, and the asteriks represent where the pointer from args will go to.

Address	Type	Name	Value	
0x00A	IntPoint2D	point_a	unknown	
0x00B	String[]	args	Received from	

			caller	
0x00C				
0x00D				
0x00E				
0x00F				
0x00G				
0x00H				
0x00I			*****	

Part 3: Now, the memory value of our Intpoint2D space will change to the memory address where the first field of our object is stored. We can now fill in the rest of our diagram with the values of our object.

Address	Type	Name	Value	
0x00A	IntPoint2D	point_a	0x00G *****	
0x00B	String[]	args	Received from caller *****	
0x00C				
0x00D				
0x00E				
0x00F				
0x00G	int	_X	1 *****	
0x00H	int	_Y	5	
0x00I			*****	

This diagram is assuming that we have created one instance of an IntPoint2D object, if we were to create more, then our diagram would have to be much larger. In this case, however, our diagram is complete. **** I tried to make an arrow as a pointer but it wasnt working,so just imagine that the asterisks represent some sort of arrow***