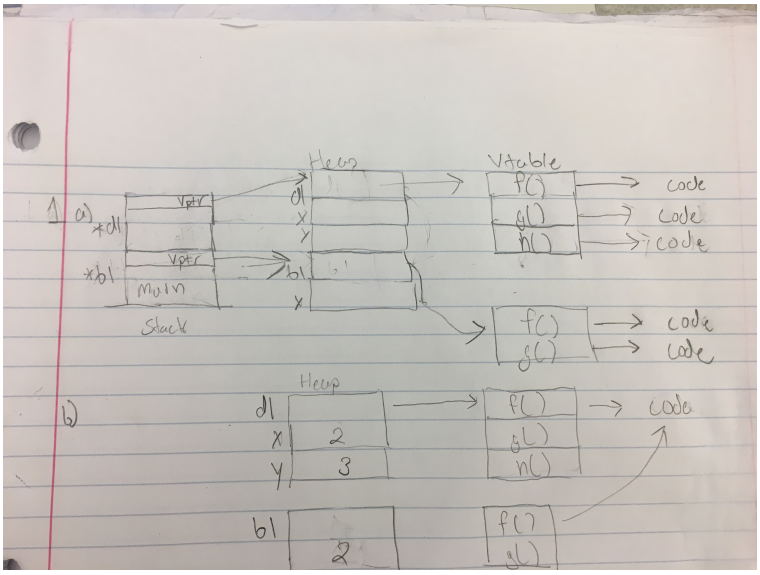
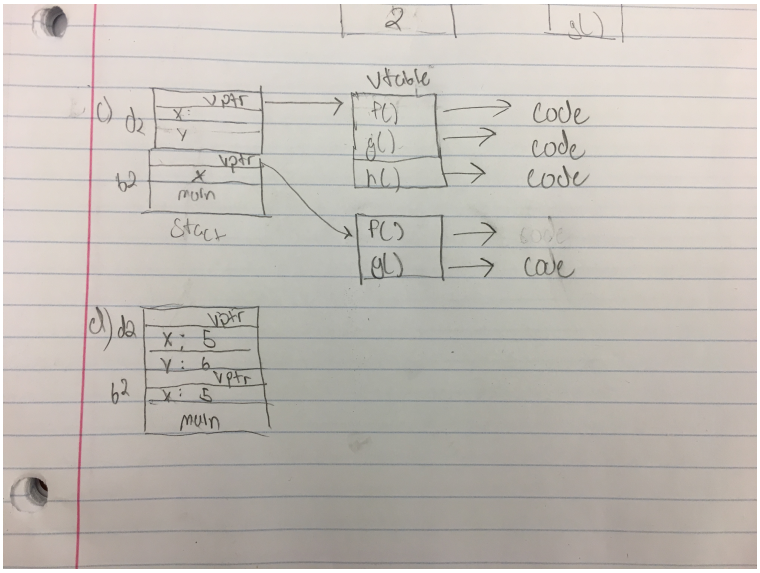
1. (a)

(b)



(c)

(d)



(e) It's because b2 is still of class vehicle which does not possess a y variable. Therefore, it only reassigns its x value.

(f) While it may be equated to d2, it's still of type airplane so the compiler will not consider b2 to be of type airplane when it is clearly vehicle, despite it being a super class.

(g) Since b2 is on the stack, it can call g() directly. Meanwhile, b1 is a pointer on the stack. Therefore, it must point to an object that points to a vtable that calls the method g().

(h) It is because g() is not designated as a virtual method by the virtual keyword. The keyword allows the method to be overridden in subclasses. Without it, it cannot be overwritten by a subclass, only inherited.

(i) This causes the class Airplane to no longer be a subtype of Vehicle. Therefore when the statement b1 = d1, is evaluated, we get and error since they are no longer subtypes.

2. (a) In file q2.cpp.

(b) It's because, in order for template functions to work, they need to know the size of the datum so that addressing into the activation record. The string literals `"Oh "` and `"noes!"` are both rvalues which don't have locations in memory and have sizes that cannot be predicted since they are essentially character arrays. Therefore, the template function cannot discern the size of it and we get a compiler error.