1. $\forall$ memberFunction $\in$ C.locallyDefinedMemberFunctions,

   memberFunction.name $\neq$ F.name.

   (the function is not already defined locally)

2. $\forall$ F2 $\in$ inheritedMemberFunctions(C),

   (F2.name = F.name) $\Rightarrow$

   matchingSignatureP (F, F2).

   (the signature matches that of any inherited function with the same name)

3. $\forall$ F3 $\in$ functionsThatOverride(F),

   matchingSignatureP(F, F3).

   (the signatures of corresponding functions in subclasses match it)

4. $\forall$ F2 $\in$ inheritedMemberFunctions(C),

   (F2.name = F.name) $\Rightarrow$

   ($\forall$ class $\in$ C $\cup$ subclassesOf(C),

   unrefdOnInstancesP(F2, class)) $\vee$

   (semanticallyEquivalantP F, F2).

   (if there is an inherited function with the same name, either the inherited function is unreferenced on instances of C (and its subclasses), or the new function is semantically equivalent to the function it replaces)

5. compilesP(F, C).

   (F will compile as a member of C.)