Exercise on a standard proof technique and effective enumerability

1. Using the technique of the proof of weak completeness, prove that the following sentence is a theorem or find a counterexample in the domain of natural numbers:

$$((x)Fx \;\&\; (\exists x)Gx) \rightarrow (\exists x)(Fx \;\&\; Gx)$$

2. Do the same for the following sentence:

$$(x)(Fx \vee Gx) \rightarrow ((x)Fx \vee (x)Gx)$$

3. Our proof of Lemma Four of Henkin's proof suggests how a list of all of the sentences of our formal language $\mathcal{L}$ might be constructed. Given such a list and given the standard proof technique illustrated in the two foregoing problems, try to devise *a step-by-step procedure* for generating a list of all the *valid* sentences of $\mathcal{L}$. One problem in devising such a procedure is that our standard proof technique regularly generates an unending sequence of sentences. Suppose that this appears to be happening when we apply our standard proof technique to the very first sentence on our list of sentences: the sequence generated by the technique has grown to over a million sentences with no inconsistency appearing *so far*. How do we ever advance to the second sentence on our list of sentences without the risk of missing one? (Hint: suppose we have a computer program for our standard proof technique and an unlimited number of computers.)