Math 69: Logic	**Winter '23**

Homework assigned January 13, 2023

Prof. Marcia Groszek | ***Student: Amittai Siavava***

Credit Statement

I worked on these problems alone, with reference to class notes and the following books:

(a) **A Mathematical Introduction to Logic** by **Herbert Enderton**.

Problem 8.

Prove Theorem 17F: *A set of expressions is decidable iff both it and its complement (relative to the set of all expressions) are effectively enumerable.*

Remark: Two semidecision procedures make a whole.

Let Σ be a set of expressions, with Γ as its complement.

We first prove that if Σ is decidable then both Σ and Γ are effectively enumerable:
Suppose Σ is decidable, then we can always determine whether a wff α is in Σ or is not in Σ. Therefore, we can implement an enumeration algorithm as follows:

(i) Pick an arbitrary wff, β, that has not yet been listed as a member of Σ or β.
(ii) If $\beta \in \Sigma$, then list it as an element of Σ.
(iii) However, if $\beta \notin \Sigma$, then list it as an element of Γ.
(iv) Repeat from step 1.

We then prove that if both Σ and its complement, Γ, are effectively enumerable then Σ is decidable. Assume that both Σ and Γ are effectively enumerable. Then, by definition of effectively enumeration, we may list members of Σ and non-members of Σ (i.e. members of Γ), and every member or non-member will eventually be listed in the appropriate category even if the enumeration might never end in the case of an infinite Σ or Γ. Consequently, by checking the listed wffs we may always determine whether a wff β is in Σ or not in Σ, implying that Σ is decidable.

Problem 10.

Let Σ be an effectively enumerable set of wffs. Assume that for each wff τ, either $\Sigma \vDash \tau$ or $\Sigma \vDash \neg\tau$. Show that the set of tautological consequences of Σ is decidable.

(a) Do this where "or" is interpreted in the exclusive sense: either $\Sigma \vDash \tau$ or $\Sigma \vDash \neg\tau$, but not both.

The compactness theorem tells us that if $\Sigma \vDash \tau$, then there exists a finite subset Σ_0 such that $\Sigma_0 \vDash \{\tau\}$.

Since Σ is effectively enumerable, we can create an algorithm to list members of Σ, where σ_k is the kth member of Σ to be listed. We can then generate finite subsets of Σ by taking an increasing set of the listed elements, say

$$\Sigma_i = \begin{cases} \varnothing & \text{if } i = 0. \\ \Sigma_{i-1} \cup \{\sigma_i\} & \text{otherwise.} \end{cases}$$

For any arbitrary wff τ, since either $\Sigma \vDash \tau$ or $\Sigma \vDash \neg\tau$, there *must* be some finite $\Sigma_k \subseteq \Sigma$ such that either $\Sigma_k \vDash \{\tau\}$ or $\Sigma_k \vDash \{\neg\tau\}$. When we find such a subset, we can mark τ as a tautological consequence of Σ or mark $\neg\tau$ as a tautological consequence of Σ. If the condition does not yet hold, we can keep growing our subset.

(b) Do this where "or" is interpreted in the inclusive sense: either $\Sigma \vDash \tau$ or $\Sigma \vDash \neg\tau$, or both.

There are two scenarios: the exclusive case and the inclusive case. In the exclusive case, we can proceed as in part a. In the inclusive case, the compactness theorem tells us that:

(i) There must exist a finite $\Sigma_1 \subseteq \Sigma$ such that $\Sigma_1 \vDash \tau$.

(ii) There must also exist a finite $\Sigma_2 \subseteq \Sigma$ such that $\Sigma_2 \vDash \neg\tau$.

Take $\Sigma_3 = \Sigma_1 \cup \Sigma_2 \subseteq \Sigma$. Then $\Sigma_3 \vDash \{\tau\}$ and $\Sigma_3 \vDash \{\neg\tau\}$.

Since no truth assignment may assign both τ and $\neg\tau$, Σ is not satisfiable and we may infer anything from the set Σ.

Problem 11.

(b) Explain why the intersection of two effectively enumerable sets is again effectively enumerable.

Suppose A and B are two effectively enumerable sets, then:

(i) We can list members of A (by definition of effective enumeration), and every member of A will eventually be listed.

(ii) We can list members of B in the same manner.

Since all members of $A \cap B$ are members of both A and B, we can list members of $A \cap B$ by listing members of A, listing members of B, and comparing which elements show up in both listings.