What to Know for the Second Exam on 12/14/2017

The exam will consist of short answer questions and proofs. The proofs will all be informal, metatheoretic proofs about our systems rather than formal proofs within the system. The balance of points will be less heavily tilted toward proofs than what was the case on the first exam.

Although you won't be explicitly tested on the material, you should remember the material from earlier in the term that have been relevant for the material in the second half of the class, including the notions of semantic consequence and the proof rules of first-order logic.

I refer to some of Smith's theorems by number on this sheet, but I do not expect you to memorize them this way. For instance, it would be out of bounds for me to ask, "State Theorem 4.2."!

For Exam 2, you should know:

• What an inductive definition is: what its parts are (base clause, inductive clause(s), closure or final clause) and how to construct one. • What an inductive proof is, what its parts are (basis step, inductive hypothesis, inductive step(s)), and how to carry one out. • The core concepts of set theory: the empty set, membership in a set, subset, powerset, and cardinality. • Frege's Basic Law V and Russell's paradox. • The Powerset Axiom and the Axiom of Infinity.

• What countability means, and how to prove that a set is countable. • The different types of cardinality, including the facts we covered about the sequence of infinite cardinalities ($\aleph_0$, $\aleph_1$, ...). • What the (generalized) continuum hypothesis says. • What the Löwenheim-Skolem theorem says, and a rough idea of how we know that it's true. • The Compactness Theorem: what it says and how to prove it.

• What a function is, including the terminology we use to classify them (total, partial, injective, surjective, bijective). • Composition of functions. • The relation between functions and properties and sets, including what a characteristic function is.

• Diagonalization proofs (e.g, Cantor's proof that $|\mathbb{N}| < |\mathbb{R}|$ or the proof that the powerset of $\mathbb{N}$ is uncountable.) • The Gödelian Puzzle (Smullyan's thought experiment involving the computer that never prints a false sentence).

• What an effectively computable function is, and how to prove that a function is effectively computable. • What an effectively enumerable set is, and how to prove that a set is effectively enumerable. • What an algorithm is, and what its numerical domain is. • What the complement of a set is. • What an effectively decidable set or property is, and how to prove one is effectively decidable. • The relationship between these different notions, as expressed (and proved) in the theorems from Chapter 3 of Smith. • Theorem 3.8 (The Basic Theorem of Arithmetic) and its proof.

• The core concepts related to theories of arithmetic: what the parts of a theory are, what axioms are, what makes a theory count as negation-incomplete, etc. • What negation completeness and consistency mean. • What it is for a theory to decide a sentence. • What it is for a theory itself to be effectively decidable. • What it is for a theory to be effectively axiomatized. • The language of arithmetic, including its syntax and its standard interpretation.

• Expressing a property; capturing a property; the difference between the two. • Weak capturing; strong capturing; and the difference between those and simply capturing a property.

• What a sufficiently expressive language is. • The theorems about sufficiently expressive languages and their proofs (Theorems 6.1–6.3). What a sufficiently strong theory is, and the theorems about those theories along with their proofs (Theorems 7.1 & 7.2).

• The systems of BA (Baby Arithmetic), Q (Robinson Arithmetic), $\mathsf{I}\Delta_0$, $\mathrm{I}\Sigma_1$, and PA (Peano Arithmetic). Know their absolute strengths (what sorts of things they can and cannot prove) and their relative strengths (what one can prove that another cannot). • What a counter-model is, and what the counter-model we used for proving Theorem 10.8 shows.

• Bounded Quantifiers • The classification of sentences based on their quantifier complexity: What a Δ_0 wff is, what a Σ_1 wff is, and what a Π_1 wff is. • Whether and why Q is Δ_0-Complete, Σ_1-Complete, and/or Π_1-Complete. • Goldbach's conjecture and the consequences of consistency with Q. • Induction schema and axioms. • Presburger arithmetic: what it is and its curious feature.

• Primitive Recursion. • The initial functions (Successor, Zero, Identity). • Defining functions by composition or recursion. • The extensionality of functions (Chapter 14.3). • The relationship between effective computability and primitive recursiveness. • The relationship between primitive recursiveness and for-loop computability. • Theorem 15.1 and its proof.

• The β-function: what it's for, what it does, what its arguments/inputs are, and what it returns, and what its quantifier complexity is. (You don't need to memorize Gödel's mathematical formula itself). • Using the β-function to prove the hard clause of Theorem 15.1.

• Theorem 16.1 • p.r. adequacy. • Theorem 17.2 and its proof.

• The basic techniques of arithmetizing syntax using Godel numbering. What a Godel number (g.n.) is. What a Super Godel number (s.g.n.) is. You don't need to memorize the codes for the symbols, but if you're provided with a code, you should know how to encode or decode g.n.'s and s.g.n.'s using prime factorization.

• What the diagonalization of a wff is. • The $Gdl(m,n)$ relation. That it's p.r. and that it can be captured in a Σ_1 wff of L_A—$\mathsf{Gdl}(\overline{m},\overline{n})$) • The construction of the canonical Gödel sentence G for PA, and its quantifier complexity. • The proof that G is true iff it's unprovable in PA. • Theorem 21.2, Theorem 21.3, and their proofs.

• The syntactic version of the Incompleteness for PA, Part (A): (If PA is consistent, then $\mathsf{PA} \nvdash \mathsf{G}$) and its proof. • The ω concepts: ω-(in)completeness, ω-inconsistency. • The relationship between ω-consistency and consistency (which is stronger)? • Syntactic Incompleteness for PA, Part (B): If PA is ω-consistent, then $\mathsf{PA} \nvdash \neg\mathsf{G}$. • Incomplete*ness* versus Incomplet*ability* • How the Incompleteness results for PA generalize to other arithmetic theories. • "Nice" theories.

• The Prf_T property and Prf_T wff. • The Prov_T wff. What's its quantifier complexity, and why? Is it p.r.? Does it *express* theoremhood in theory T? Why, or why not? • Theorem 24.1 and its proof. • "$\mathsf{T} \vdash \mathsf{G}_T \leftrightarrow \neg\mathsf{Prov}_T(\ulcorner \mathsf{G}_T \urcorner)$" • The Diagonalization Lemma (Theorem 24.4) and an understanding of its proof. • How to use the diagonalization lemma to prove Incompleteness again. • The proof that no nice theory can capture the Prf_T property (theoremhood).

The End!