University of Nevada, Las Vegas Computer Science 456/656 Fall 2026

Study Guide for Third Examination Tuesday November 24, 2026

First Draft so you can get started.

1. True/False/Open
   (i) _____ The language of all regular expressions over the alphabet $\{a, b\}$ is a regular language.
   (ii) _____ Every context-free language is $\mathcal{NC}$.
   (iii) _____ Co-$\mathcal{P}$–SPACE = $\mathcal{P}$–SPACE.
   (iv) _____ Every subset of a regular language is regular.
   (v) _____ Every subset of a $\mathcal{RE}$ language is regular.
   (vi) _____ Every subset of a countable set is countable.
   (vii) _____ Every subset of an uncountable set is uncountable.
   (viii) _____ Any Turing machine can be emulated using C++.
   (ix) _____ All sliding block problems are $\mathcal{P}$–TIME
   (x) _____ All sliding block problems are $\mathcal{P}$–SPACE
   (xi) _____ If L_1 and L_2 are languages, L_2 is $\mathcal{NP}$–complete, and there is a polynomial time reduction of L_1 to L_2, then L_1 must be $\mathcal{NP}$–hard.
   (xii) _____ If L_1 and L_2 are languages, L_1 is undecidable, and there is a recursive reduction of L_1 to L_2, then L_2 must be undecidable.
   (xiii) _____
   (xiv) _____
   (xv) _____
   (xvi) _____
2. Know the definitions of
   (i) Certificate, in the context of $\mathcal{NP}$,
   (ii) Reduction.
   (iii) Countable (Enumerabl).
   (iv) Recursively Enumerable.
   (v) $\mathcal{NP}$-completeness.
   (vi) SAT.
   (vii) 3-SAT.
   (viii) $\mathcal{P}$-SPACE.
   (ix) Independent Set problem?
   (x) Nick's Class.
   (xi) $\mathcal{P}$-completeness,

(xii) CVP.

(xiii) recursive real number.

(xiv) Turing machine.

(xv) Turing power.

(xvi) Context-sensitive grammar.

(xvii) General (or Unrestricted, or Phase-Structure) grammar.

(xviii) The power set of a set.

3. Know the $\mathcal{P}$–TIME reductions of

   (i) The subset sum problem to the partition problem.

   (ii) 3-SAT to the independent set problem.

4. What of these problems/languages are known to be $\mathcal{NC}$?

   (i) Arithmetic matrix multiplication.

   (ii) Boolean matrix multiplication.

   (iii) Tropical matrix multiplication.

   (iv) CVP.

   (v) The Dijk language.

   (vi) Computing the sum of a list of numerals.

   (vii)

5. What of these problems/languages are known to be $\mathcal{NP}$-complete?

   (i) 2-SAT

   (ii) 3-SAT

   (iii) 4-SAT

   (iv) TSP

   (v) Deciding whether the red car can escape a given configuration of RUSH HOUR.

   (vi) The Dijk language.

   (vii) The binary numeral factoring problem.

   (viii)

6. Why is the problem of whether $\mathcal{NC} = \mathcal{P}$ so importance nowadays, as opposed to in the past?

7. What is the Church-Turing thesis, and why is it important?

8. Know the proofs of these theorems.

   (i) A language L is decidable if and only if some program enumerates L in canonical order.

   (ii) A language L is accepted by some machine if and only if some program enumerates L.

   (iii) The halting problem is undecidable.

   (iv) The set of real numbers is uncountable.

   (v) Use the pumping lemma to prove that $L = \{a^n b^n \ : \ n \geq 0\}$ (or some other simple example) is not regular,