CS747 - RANDOMIZED METHODS IN COMPUTATIONAL COMPLEXITY
NITIN SAXENA

ASSIGNMENT 1

POINTS: 50

DATE GIVEN: 12-AUG-2025 DUE: 03-SEP-2025

Rules:
- You are strongly encouraged to work *independently*. That is the best way to understand & master the subject.
- Write the solutions on your own and honorably *acknowledge* the sources if any. `http://cse.iitk.ac.in/pages/AntiCheatingPolicy.html`
- Submit your solutions, before time, to your TA. Please give your LaTeXed or Word processed solution-sheet in PDF. This will be graded, and commented, in-place.
- Clearly express the fundamental *idea* of your proof/ algorithm before going into the other proof details. The distribution of partial marks is according to the proof steps.
- There will be a penalty if you write unnecessary or unrelated details in your solution. Also, do not repeat the proofs done in the class.
- Problems marked '0 points' are for practice.
- Acknowledgements: Several problems are from *Arora & Barak, Computational Complexity: A Modern Approach*, and other lecture notes.

Question 1: [NP-c] *[5 points]* Let $\mathbb{F}_p$ be a prime field. Show that the question of existence of zeros, of a system of *quadratic equations*, is NP-complete.

Question 2: [AC^0] *[10 points]* Consider the question of adding two n-bit numbers. Show that it can be done by a poly(n)-sized, *constant-depth boolean* circuit.

[This result is usually stated as $Addition \in AC^0$.]

Question 3: [QBF] *[15 points]* Recall the problem of testing the truth of a quantified boolean formula (QBF). Show that QBF is PSPACE-complete.

Question 4: [Amplification] *[10 points]* In the definition of BPP we had used an error probability of 1/3 (or 1/4). Show that the class BPP remains the same if we increase the error-probability upper bound to $\frac{1}{2} - \frac{1}{\text{poly}(n)}$, where n is the input size.

Question 5: [Non-uniform Derandomization] *[10 points]* Show that BPP $\subseteq$ P/poly.

Question 6: [Turing vs. Circuits] *[0 points]* Consider the circuit-complexity class Size(n). Show that it has *uncomputable* problems.

Question 7: [Permanent] *[0 points]* The question of *counting* the number of satisfying assignments of a given boolean formula is called #SAT. Show that #SAT and permanent (for 0/1 matrices) are poly-time equivalent functional problems.

Question 8: [Time hierarchy] *[0 points]* Let $s(n)$ be a real-valued polynomial. Prove that Dtime($s(n)$) is a proper subset of Dtime($s(n)^2$).

Question 9: *[0 points]* State and prove the *hierarchy theorems* for Ntime($s(n)$) and Space($s(n)$).

Question 10: *[0 points]* In Q.4. if we further increase the error-probability (upper bound) to $\frac{1}{2} - \frac{1}{2^n}$, what complexity class do you get? Could this be said to be capturing *efficient* randomized algorithms?

Question 11: *[0 points]* Let $d \in \mathbb{N}$ and a prime p be given in the input in binary. Give a poly($d \log p$)-time randomized algorithm to construct the finite field $\mathbb{F}_{p^d}$.

Question 12: *[0 points]* Over the field of rationals, an *algebraic* circuit of size s may compute numbers of magnitude 2^{2^s}. It's impossible to work with exponential-bits in *poly*-time. Is there a practical way to solve *Polynomial Identity Testing* in this case?

Question 13: [Circuit simulation] *[0 points]* Consider the circuit complexity class Size($n^{O(1)}$). Could you redefine it so that its problems can be simulated on poly-time Turing machines?

Question 14: [Uncomputable] *[0 points]* Make a list of problems that are *provably* uncomputable by machines.

Question 15: [Field extension] *[0 points]* Classify the field extensions that lie above $\mathbb{C}$ resp. $\overline{\mathbb{F}}_p$.

□□□