

ASSIGNMENT 4

POINTS: 50

DATE GIVEN: 25-OCT-2025

DUE: 13-NOV-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: [9 points] Suppose a boolean function f is in E with $H_{\text{avg}}(f) \geq n^4$. Show that the function $g : z \mapsto z \circ f(z)$, for $z \in \{0, 1\}^\ell$, is an $(\ell + 1)$ -prg.

Question 2: [13 points] Prove that $\text{NEXP} = \text{MA}$ implies $\text{NEXP} \subseteq \text{P/poly}$.

Question 3: [10 points] Suppose $\text{BPP} \neq \text{EXP}$. Could you use this to derandomize BPP to some extent? Sketch the proof details.

Question 4: [9 points] For every $\delta > 0$ and sufficiently large n , prove the existence of a *linear* ecc $E : \{0, 1\}^n \rightarrow \{0, 1\}^{1.1n/(1-H(\delta))}$ with distance at least δ .

(Note: *Linear code* means that $E(x) + E(y) = E(x + y)$, where

addition is componentwise modulo 2. Let $H(\delta) := -\delta \log \delta - (1 - \delta) \log(1 - \delta)$.)

Question 5: [9 points] Show that there exists an *explicit linear-stretch* code, i.e. $\exists c, \delta > 0, \forall n, \exists \text{ ecc } E : \{0, 1\}^n \rightarrow \{0, 1\}^{cn}$ of distance at least δ with efficient encoding/decoding algorithms.

Question 6: [0 points] For every $c > 0$, prove that $\text{EXP} \not\subseteq \text{i.o.-Size}(n^c)$.

Question 7: [0 points] In the lectures we've been using multiple versions of the Permanent function per_n of an $n \times n$ matrix. It could be boolean-valued, integral, or algebraic/ arithmetic. Carefully consider these definitions and compare them.

Question 8: [0 points] Prove that, for unique decoding, the channel error should be less than 25%. What about *non*-unique decoding?

Question 9: [0 points] How do you factor $f(x) \bmod p$?

Question 10: [0 points] How do you factor $f(x_1, x_2) \bmod p$?

Question 11: [0 points] How do you find an integral root of an *integral* polynomial $f(x)$?

Question 12: [0 points] Why does Reed-Solomon code seem to 'violate' the Johnson Bound on list-decoding?

Question 13: [0 points] Complete the technical details of the amplification of worst-case hardness to average-case hardness.

Question 14: [0 points] Give an *upper* bound on the average-case hardness of a boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$. E.g. could it be *super*-exponential?

Question 15: [0 points] Recall the definition of (S, ε) -pseudorandom distribution. What happens to the theory of prg if we take $\varepsilon = 1/2$?
What happens if we take $\varepsilon = 1/2 - 1/S$?

Question 16: [0 points] How do you test two integral polynomials $f(x), g(x) \bmod p^2$ for *co-primality*? For irreducibility?

□□□