This is a *sample question paper*, resembling the format of the upcoming Mid-Semester Examination. One sample problem is provided to indicate the typical style of questions to appear in the final paper. Note that you will have to answer *two-out-of-three* such questions, each carrying 25 marks, in the examination. In addition, there will be a single bonus problem, worth 10 marks, for you to attempt.

INDIAN STATISTICAL INSTITUTE

Mid-Semester of Second Semester Examination : 2015–16

Course : Bachelor of Statistics (Hons.)

Subject : Numerical Analysis : BStat-I

Date : 22 February 2016 | Maximum Marks : 60 | Duration : 3 Hours

---

Problem 1 **[25]**

Given a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, the *simple LU decomposition* of $\mathbf{A}$ may be defined as $\mathbf{A} = \mathbf{LU}$, where $\mathbf{L} \in \mathbb{R}^{n \times n}$ is a lower-triangular matrix, and $\mathbf{R} \in \mathbb{R}^{n \times n}$ is an upper-triangular matrix.

A. Describe an algorithm that may decompose $\mathbf{A}$ into $\mathbf{L}$ and $\mathbf{U}$ through successive left multiplication by lower triangular matrices, such that $\mathbf{L}_n \cdots \mathbf{L}_2 \mathbf{L}_1 \mathbf{A} = \mathbf{U}$, and $\mathbf{L} = (\mathbf{L}_n \cdots \mathbf{L}_2 \mathbf{L}_1)^{-1}$. [8]

B. Is it always possible to obtain such a *simple LU decomposition* of $\mathbf{A} \in \mathbb{R}^{n \times n}$?

- If so, provide a complete justification that your proposed algorithm (in Part A) achieves a *simple LU decomposition* for any matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$.
- If not, state a case where a *simple LU decomposition* may not be feasible. How would you modify the proposed algorithm (in Part A) in such a case?

[7]

C. If it is possible to perform the *simple LU decomposition* for some $\mathbf{A} \in \mathbb{R}^{n \times n}$, describe an algorithm that would perform the decomposition "in place", that is, store the decomposed components $\mathbf{L}$ and $\mathbf{U}$ by overwriting the actual matrix $\mathbf{A}$. [10]