Corollaries to VIT - 1

- **Corollary 1.3**: Suppose a square matrix A is factored as a product of square matrices, i.e. $A = A_1A_2......A_n$ with $n \geq 2$. Then A is invertible if and only if each A_i is invertible.
  - **Note:** The above Corollary 1.3 applies only if the matrices A_i are ***all square matrices***. It is certainly possible to factor a square matrix as a product of rectangular matrices. The above result cannot be applied to such a factorization - non-square matrices are never invertible. ***The backward direction, i.e. that if each A_i is invertible, then so is their product, was noted in the Quick Review at the beginning. So we only need to show that if A is invertible, then so is each A_i.***

Uniqueness of RREF Matrix

- **Corollary 1.5:** The RREF matrix of any given matrix is unique, i.e. a matrix cannot be row-equivalent to two distinct RREF matrices. Alternatively, two distinct RREF matrices cannot be row-equivalent to each other.
- **Remark 1**: We had earlier stated the above as a remark: *Remark 2 after Proposition 2*. It can be proved using VIT, and is therefore formally stated as a corollary.
- **Remark 2**: However, the proof is difficult and so it is optional: ***you are not required to be familaiar with the proof.*** A proof is presented in Appendix A of the textbook (David C. Lay), which you may refer to.
- ***If students are interested, proofs of Corollaries 1.3 and 1.5 will be presented in an extra (optional) session.***

LU Factorization of a Matrix - 1

- **Background:** We have briefly looked at factorization of matrices. There are a few specific ways to factorize matrices which are useful, usually because they are more efficient for solving problems using computers. We will study the LU factorization at present because it is very closely related to the row-reduction algorithm.
- **Motivation/Application:** A problem which occurs frequently in applications is to solve equations of the form $A\mathbf{x} = \mathbf{b}_1$, $A\mathbf{x} = \mathbf{b}_2$, …., $A\mathbf{x} = \mathbf{b}_p$ where the $\mathbf{b}_j$ change but the coefficient matrix A remains fixed. Often, the matrix A is invertible, so one way could be to calculate A^{-1} and then obtain the solutions $A^{-1}\mathbf{b}_j$. However, it frequently turns out to be more efficient to calculate an LU factorization of A, which requires reducing to an echelon form only. The equations are then solved using the LU factorization.

LU Factorization of a Matrix - 2

- **Definition:** Suppose that A is an $m \times n$ matrix which can be reduced to an echelon form matrix ***without using row-interchange operations***. That is to say, only replacement operations are used in the forward phase of the row reduction algorithm. Then A can be factorized as A = LU, **where L is an $m \times m$ lower triangular matrix with 1's on the diagonal, and U is an $m \times n$ echelon form matrix obtained from A by row reduction.** Any such factorization is called an LU factorization of A. The matrix L is invertible and is called a **unit lower triangular matrix.**
- **Remarks:**
  - The definition is given for rectangular matrices, but in practice this approach is used mainly for square matrices.
  - The matrix A need not be invertible, but again the approach is useful mainly for invertible matrices.
  - There is a generalized version of this definition for the case when row reduction of the matrix to echelon form requires row interchange operations. The general case will be briefly explained later, without going into details.
-

LU Factorization of a Matrix - 3

- **Application to Solving Linear Systems:** Before describing th algorithm for constructing an LU factorization of a matrix A, we indicate how the LU factorization is used to solve a linear system.
- **Consider:**

  $$\mathrm{A}\mathbf{x} = \mathbf{b} \Rightarrow (\mathrm{LU})\mathbf{x} = \mathbf{b} \Rightarrow \mathrm{U}\mathbf{x} = \mathrm{L}^{-1}\mathbf{b}$$

  Note that the final step above is legitimate since L is known to be invertible.

  Now, $\mathrm{L}^{-1}\mathbf{b}$ is the solution of the system $\mathrm{L}\mathbf{y} = \mathbf{b}$

  So, solution of the original system $\mathrm{A}\mathbf{x} = \mathbf{b}$, is replaced by the solution of two systems: $\mathrm{L}\mathbf{y} = \mathbf{b}$ and then $\mathrm{U}\mathbf{x} = \mathbf{y}$
- **Remark:** This works only because the two systems are triangular and so very simple and easy to solve. We will show this with an example.

LU Algorithm - 1

- Input an $m \times n$ matrix A
- **Step 1**: Row reduce A, if possible, to an echelon form matrix U, using only row replacement operations that ***add a multiple of a row to a row below it.***
- **Step 2:** Place entries in L ***such that the same sequence of row operations reduces L to I.***
- **Remark:** a) The computational efficiency of the LU approach depends on the fact that L is obtained without doing any significant extra work. This will be shown in the example.

  b) Step 1 is not always possible. But if it is, then the theoretical justification on the next slide (after the example) indicates why an LU factorization is then obtained.

LU Algorithm - 2

- **Theoretical Justification:**
- Suppose it is possible to row reduce A to an echelon form matrix U using only row replacement operations that ***add a multiple of a row to a row below it.***
- Then there are unit lower triangular elementary matrices E_1, E_2, …., E_p such that: $E_p \ldots\ldots E_1 A = U$.
- Hence $A = (E_p \ldots\ldots E_1)^{-1} U = LU$ where $L = (E_p \ldots\ldots E_1)^{-1}$ is clearly invertible. Furthermore, the inverses and products of unit lower triangular matrices are also unit lower triangular matrices (*left as an exercise !*).
- Finally $L = (E_p \ldots\ldots E_1)^{-1} I$ and so $I = (E_p \ldots\ldots E_1)L$, i.e. the same sequence of row operations that reduces A to U reduces L to I.

LU Factorization – General Case

- **Concluding Remarks:** As we can easily see with an example, sometimes row interchanges are needed if we want to get an **echelon form** matrix U from an invertible matrix A. Furthermore, in working with large systems, row interchanges are nearly always used for computational stability. This general situation can be handled in nearly the same way, except that the resultant L is not necessarily unit lower triangular, but is **permutated unit lower triangular**, i.e. we can make L into a unit lower triangular matrix by a permutation of the rows. Such a factorization is also usually referred to as an LU factorization.
- **Note:** ***We will not consider the general case in this course.*** It would usually be covered in an advanced or specialized course on numerical linear algebra or computational linear algebra