# Exploring Linear Algebra

## Labs and Projects with MATLAB®

## Textbooks in Mathematics

Series editors:
Al Boggess and Ken Rosen

APPLIED FUNCTIONAL ANALYSIS, THIRD EDITION
*J. Tinsley Oden and Leszek Demkowicz*

AN INTRODUCTION TO NUMBER THEORY WITH CRYPTOGRAPHY, SECOND EDITION
*James R. Kraft and Lawrence Washington*

MATHEMATICAL MODELING: BRANCHING BEYOND CALCULUS
*Crista Arangala, Nicolas S. Luke, and Karen A. Yokley*

ELEMENTARY DIFFERENTIAL EQUATIONS, SECOND EDITION
*Charles Roberts*

ELEMENTARY INTRODUCTION TO THE LEBESGUE INTEGRAL
*Steven G. Krantz*

LINEAR METHODS FOR THE LIBERAL ARTS
*David Hecker and Stephen Andrilli*

CRYPTOGRAPHY: THEORY AND PRACTICE, FOURTH EDITION
*Douglas R. Stinson and Maura B. Paterson*

DISCRETE MATHEMATICS WITH DUCKS, SECOND EDITION
*sarah-marie belcastro*

BUSINESS PROCESS MODELING, SIMULATION AND DESIGN, THIRD EDITION
*Manual Laguna and Johan Marklund*

GRAPH THEORY AND ITS APPLICATIONS, THIRD EDITION
*Jonathan L. Gross, Jay Yellen, and Mark Anderson*

A FIRST COURSE IN FUZZY LOGIC, FOURTH EDITION
*Hung T. Nguyen, Carol L. Walker, and Elbert A. Walker*

EXPLORING LINEAR ALGEBRA
*Crista Arangala*

# Exploring Linear Algebra

## Labs and Projects with MATLAB®

Crista Arangala

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742


CRC Press is an imprint of Taylor & Francis Group, an Informa business



Printed on acid-free paper
Version Date: 20190117

International Standard Book Number-13: 978-1-138-06351-8 (Hardback)
International Standard Book Number-13: 978-1-138-06349-5 (Paperback)



**Library of Congress Cataloging-in-Publication Data**

Names: Arangala, Crista, author.
Title: Exploring linear algebra : labs and projects with Matlab / Crista Arangala.
Description: Boca Raton : CRC Press, Taylor & Francis Group, 2019. | Includes bibliographical references and index.
Identifiers: LCCN 2018054578 | ISBN 9781138063495
Subjects: LCSH: Algebras, Linear--Computer-assisted instruction. | MATLAB.
Classification: LCC QA185.C65 A73 2019 | DDC 512/.5028553--dc23
LC record available at https://lccn.loc.gov/2018054578

**Visit the Taylor & Francis Web site at**
**http://www.taylorandfrancis.com**

**and the CRC Press Web site at**
**http://www.crcpress.com**

# Contents

# Preface

This text is meant to be a hands-on lab manual that can be used in class every day to guide the exploration of linear algebra. Most lab exercises consist of two separate sections, explanations of material with integrated exercises, and theorems and problems.

The exercise sections integrate problems, technology (MATLAB R2017b), MATLAB visualization, and MATLAB simulations that allow students to discover the theory and applications of linear algebra in a meaningful and memorable way. It is important to note that on a very few occasions, the Symbolize Toolbox features that are included in MATLAB R2017b, and not in previous versions, are implemented.

The intention of the theorems and problems section is to integrate the theoretical aspects of linear algebra into the classroom. Instructors are encouraged to have students discover the truth of each of the theorems and proofs, to help their students move toward proving (or disproving) each statement, and to allow class time for students to present their results to their peers. If this course is also serving as an introduction to proofs, we encourage the professor to introduce proof techniques early on as the theorem and problems sections begin in Lab 3.

There are a total of 80 theorems and problems introduced throughout the labs. The author has intentionally labeled those results that are traditional linear algebra theorems as theorems in these sections and has labeled other significant results and interesting problems as problems. There are, of course, many more results, and users are encouraged to make conjectures followed by proofs throughout the course.

In addition, each chapter contains a project set that consists of application-driven projects that emphasize the material in the chapter. Some of these projects are extended in follow-up chapters, and students should be encouraged to use many of these projects as the basis for further undergraduate research.

# *Acknowledgments*

Each time I publish a book, my father, Joseph Coles, jokingly asks if I have dedicated the book to him. I have made dedications to my children, to my colleagues, and to my students, but I really would never have gotten to where I am today if my parents, Joseph and Carol Ann Coles, had not taught me to be strong and confident. So this one is for you Dad and Mom. Thanks for all your support.

The writing of this text was supported by an Elon University Funds for Excellence Grant. I would also like to thank my students in my Fall 2018 Linear Algebra class, Megan Bargstedt, Sarah Boggins, Samantha Chessen, Kasey Collins, Emily Cooper, Cecilia Dong, Matthew Foster, Michael Golaski, Eduardo Gonzalez, Hannah Noelle Griesbach, Joseph Keating, Yousaf Khan, Ryan Kugal, Carter Martin, McKenzie Miller, Amy Moore, David Norfleet, Timothy Redgrave, William Reynolds, Daniel Ryan, Isaac Sasser, Shannon Treacy, and Anne Williams, for helping me work through the manuscript before it went to publication.

# 1

# *Matrix Operations*

## Lab 0: An Introduction to MATLAB®

### Introduction

MATLAB is a computer programming language that allows for matrix manipulation. MATLAB only recognizes certain commands that are relative to this program. Therefore you must type the commands as you see them. MATLAB is also case sensitive which means that if you see uppercase you must type uppercase and if you see lowercase you must type lowercase.

There are two ways that you can effectively use a MATLAB command. One way to run a MATLAB command is to type the command directly in the Command Window next to the << symbol and then hit return. This process is convenient when processing only one command at a time makes sense. When you wish to evaluate more than one command at once, it might make more sense to open a MATLAB script.

In order to start a MATLAB script, click on New Script in the tool bar and start typing your commands. In order to process your script after typing it, save your script and then click on the run button in the tool bar. It is also important to note that if you close MATLAB and come back to your work later, your work is not stored in the memory so it is a good idea to save your work so that you can reevaluate it later.

At any point if you are having difficulties, use the Help menu; it is very helpful.

For each lab, you will have to open a new MATLAB script file, also called an Editor file, and type all solutions in this document. So let's begin there.

Open a new MATLAB script file by choosing the New drop down menu followed by the script choice.

To save this file, choose Save in the drop down menu. Save this file as lab0.m.

*Exercises:*

a. *Type:* ***x=6*** *and then press the Run button in the tool menu. Notice that the output will show up in the Command window (which is a separate window from the Editor).*

b. *On the next line in the Editor, type:* ***x=6;*** *and then press the run button. What is the difference between the output in part a and the output here? In each case, MATLAB stores 6 in the variable $x$.*

c. *Type:* ***x+5*** *and then press run.*

d. *Type:* ***disp('x+5')*** *and on the next line type* ***disp(x+5)****, then Run. Which $x + 5$ in the display statement actually produces the value 11?*

In order to comment a line out in the editor put a % at the beginning of the line. That is, a line with a % at the beginning of it will not be processed by MATLAB when it is run. It might also be important to be able to clear the entire memory or a particular variable. If you wish to clear the entire memory, type **clear** or if you wish to clear a single variable, such as $x$, type **clear x**.

## Basic Programming in MATLAB

In this section, we will assume a basic understanding of programming. We will discuss Tables of data, For Loops, and If-Then Statements here. Again, the Help menu is very helpful in this regard as well.

The colon, **:**, is one of the handiest symbols in MATLAB, as we will see. If we wish to create a Table of 11 points with values $x$, where $x$ is the integers from -1 to 9, we would type **Name of Table= -1:9;**.

To identify the $i^{\text{th}}$ entry in table type **Name of Table(i)**. To identify the entries in the $i^{\text{th}}$ row in a table type **Name of Table(i,:)**.

The structure of a For Loop is:

$$\begin{aligned}&for\ index = starting\ value : ending\ value\\ &\quad body\ statements;\\ &end\end{aligned}$$

All statements in the body of the For Loop must be separated by semicolons. We can create a table of values $x^2$ where $x$ is the integers from -1 to 9, by starting with a table of zeros, typing **Name of Table=zeros(1,11)** and then creating the for loop,

$$\begin{aligned} &for\ i = 1 : 11 \\ &\quad Name(i) = (i-2)^2; \\ &end \end{aligned}$$

*Exercises:*

a. *Create a table named Table1 with entries equal to* $4*i$*, where* $i$ *goes between 1 and 6.*

b. *Type Table1(5) in the Command Window to determine the* $5^{th}$ *entry of this table.*

c. *Type and run the following code and determine what it does.*
$A = zeros(5);$
$for\ i = 1 : 5$
$A(i,i) = 1;$
*end*
$disp(A)$

The structure of the If-Then statement in MATLAB is:

$$\begin{aligned} &if\ condition \\ &\quad body\ statements\ if\ condition\ is\ true; \\ &end \end{aligned}$$

Similarly, the structure of an If-Then-Else statement is :

$$\begin{aligned} &if\ condition \\ &\quad body\ statements\ if\ condition\ is\ true; \\ &else \\ &\quad body\ statements\ if\ condition\ is\ false; \\ &end \end{aligned}$$

Note there is also an elseif statement as well that may come in handy.

When stating conditions in your if-then statement you may have to test an equality. Here we have to distinguish in MATLAB between == and =. When you use the "=", single equals, this is an assignment where you are assigning a value. If you use the "==", double equals, MATLAB interprets this as a condition or test and returns True or False. A double equals should be used to test equality in an if-then condition.

*Exercises:*

a. *Type and run the following code and determine what it does.*
$A = zeros(5);$
$for\ j = 1:5$
$for\ i = 1:5$
$A(i,i) = 1;$
$if\ i < j$
$A(i,j) = 2;$
$end;$
$end;$
$end;$
$disp(A)$

In the above code, we call the pair of For Loops a *Nested For Loop* because one is inside the other.

b. *Write a nested for loop, with incremental variables $i$ and $j$, which incorporates an if-then statement that creates a $5 \times 5$ table, $A$, whose entries are 1 when $i = 1$ or $j = 1$. All other entries of $A$ should be zero.*

# Lab 1: Matrix Basics and Operations

## Introduction

A *matrix* is a rectangular array of numbers. The numbers in the array are called the *entries* of the matrix. $A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$ is a matrix.

$$\text{The general form is } \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix}.$$

## Defining a Matrix in MATLAB

Example: To define the matrix $A$ above, type **A=[1 2 3; 4 5 6]**.

To find the dimensions of a matrix in MATLAB,

Type: **size(The Name of the Matrix)**

*Exercises:*

a. *Define the matrix* $B = \begin{pmatrix} 3 & 4 \\ 6 & 7 \\ 9 & 10 \end{pmatrix}$.

b. *Find the dimensions of the matrices* $A$ *and* $B$.

c. *Explain what the dimensions of a matrix are telling you about the matrix.*

## Operations on Matrices

## Adding Two Matrices

To add two matrices together, type :
**The Name of the Matrix1 + The Name of the Matrix2**

*Exercises:*

a. *Find the sum* $A + B$. *You should get an error; explain why you think an error occurred.*

b. *Define matrix* $M = \begin{pmatrix} 4 & 5 & 1 \\ -1 & 3 & 2 \end{pmatrix}$. *Find* $A + M$ *and* $M + A$. *Is addition of matrices commutative?*

c. *Explain the process of matrix addition. What are the dimensions of the sum matrix. How would you take the difference of two matrices?*

## Scalar Multiplication

To multiply a matrix by a constant c,

Type : **c*The Name of the Matrix**

*Exercise: Multiply matrix A by the scalar 4. Is multiplication of a scalar from the left the same as multiplication of a scalar from the right? (i.e., does 4*A = A*4?)*

## Multiplying Two Matrices

To multiply two matrices together, type:

**The Name of the Matrix1*The Name of the Matrix2**

*Exercises:*

a. *Multiply matrix A on the right by matrix B.*

b. *Go to* `https://www.mathworks.com/matlabcentral/fileexchange/63993-matrix-multiplication-app` *and try some examples of matrix multiplication. Then describe the multiplication process.*

c. *Multiply matrix A on the left by matrix B. Was your description of the multiplication process correct? What are the dimensions of this matrix?*

d. *Multiply matrix A on the right by matrix M. You should get an error; explain why an error occurred.*

e. *Is matrix multiplication commutative? What has to be true about the dimensions of two matrices in order to multiply them together?*

## The Transpose and Trace of a Matrix

The transpose of a matrix, $A$ is denoted $A^T$. To take the transpose of a matrix,

Type : **The Name of the Matrix'**

*Exercises:*

a. *Take the transpose of matrix A and describe the transpose operation.*

b. *What are the dimensions of the matrix $A^T$?*

c. *What is $(A^T)^T$?*

d. *Calculate* $(A+M)^T$*. Does this equal* $A^T + M^T$*?*

e. *Calculate* $(AB)^T$*. Does this equal* $A^T B^T$*?*

f. *Calculate* $B^T A^T$*. What is this equal to?*

g. *Calculate* $(3A)^T$*. What is this equal to?*

h. *In the above exercises, you explored properties of the transpose of a matrix. Write down conjectures on the properties that you observed about the transpose.*

If the number of rows of a matrix is the same as the number of columns in that matrix we call the matrix a *square matrix*. The *trace* of a square matrix $A$, $tr(A)$, is a mapping taking a square matrix to a real number. To take the trace of a square matrix

Type: **trace(The Name of the Matrix)**

Define matrix $U = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 0 \\ 0 & 2 & -1 \end{pmatrix}$ and $V = \begin{pmatrix} 1 & 0 & 0 \\ 4 & 3 & 0 \\ 0 & 0 & 2 \end{pmatrix}$.

*Exercises:*

a. *Calculate* $tr(U)$ *and* $tr(V)$ *and describe the trace operation.*

b. *Calculate* $tr(U+V)$*. Does this equal* $tr(U) + tr(V)$*?*

c. *Calculate* $tr(U^T)$*. Does this equal* $tr(U)$*?*

d. *Calculate* $tr(UV)$*. Does this equal* $tr(U)tr(V)$*?*

e. *Calculate* $VU$ *and* $tr(VU)$*. Note that* $UV \neq VU$*, but does* $tr(UV) = tr(VU)$*?*

# Lab 2: A Matrix Representation of Linear Systems

## Introduction

You may remember back to the time when you were first learning algebra and your favorite math teacher challenged you to find a solution for $x$ and $y$ in a system with 2 equations with 2 unknown variables, such as $2x + 5y = 7$ and $4x + 2y = 10$. How did you do it?

My money is on solving for one variable in one equation, and substituting into the other. Or maybe you multiplied the first equation by a constant and subtracted the second from the first to solve, and then the story goes on. This method is fine and actually how we too will do it except in terms of matrices. The algorithm that we will use is called *Gaussian Elimination* (or *Gauss Jordan Elimination*).

*Exercise: How many solutions are there to a system with 2 equations and 2 unknowns (in general)? How would you visualize these solutions?*

A linear system in variables $x_1, x_2,...,x_k$ is of the form

$$\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + ... + a_{1k}x_k & = & b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2k}x_k & = & b_2 \\ a_{31}x_1 + a_{32}x_2 + ... + a_{3k}x_k & = & b_3 \\ \vdots \quad \vdots \quad \ddots \quad \ddots & & \vdots \\ a_{m1}x_1 + a_{m2}x_2 + ... + a_{mk}x_k & = & b_m \end{array}$$

and can be written as the matrix equation

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1k} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2k} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3k} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mk} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_k \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_m \end{pmatrix}.$$

In the lab below, you will find all of the terms that you will need in order to move forward with Gaussian Elimination (or Gauss Jordan Elimination).

## The Identity Matrix

The $n \times n$ *identity matrix* $I_n = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}$. This matrix has 1's down the "main diagonal" and 0's everywhere else. The command for the $n \times n$ Identity Matrix is, **eye(n)**.

## Row Echelon Form of a Matrix

A matrix is in *row echelon form* if
1) The first non-zero entry in each row is a one, called a *leading one*
2) Rows of all zeros are at the bottom of the matrix
3) All entries below leading ones are zeros
4) If $i < j$, the leading one in row $i$ is to the left of the leading one in row $j$.

In addition, the matrix is in *reduced row echelon form* if
5) each column with a leading one has only zeros everywhere else.

*Exercises:*

a. *Use MATLAB to create a* $4 \times 4$ *Identity Matrix.*

b. *Given the system* $2x+5y = 7$ *and* $4x+2y = 10$, *create a coefficient matrix, A, using the coefficients of the variables.*

c. *Find the reduced row echelon form of A, type* ***rref(A)****.*

So how do we think about getting $A$ into row echelon (Gaussian Elimination) or reduced row echelon form (Gauss Jordan Elimination)? We perform elementary row operations to the original matrix. And with every elementary row operation there is a corresponding elementary matrix.

## Elementary Row Operations and the Corresponding Elementary Matrices

There are only three possible elementary row operations.

1. **Swap two rows in a matrix.** If you swap two rows in a $2 \times 2$ matrix, start with $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and perform this operation to get elementary matrix $E_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.

2. **Multiply a row by a nonzero scalar (constant),** $k_1$**.** If you multiply

row two in a $2 \times 2$ matrix by $k_1 = -\frac{1}{8}$, start with $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and perform this operation to get elementary matrix $E_2 = \begin{pmatrix} 1 & 0 \\ 0 & -\frac{1}{8} \end{pmatrix}$.

3. **Add a nonzero multiple, $k_2$, of a row to another row.** If you add a multiple $k_2 = -2$ of row one to row two in a $2 \times 2$ matrix, start with $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and perform this operation to get elementary matrix $E_3 = \begin{pmatrix} 1 & 0 \\ -2 & 1 \end{pmatrix}$.

*Exercises: Define* $A = \begin{pmatrix} 2 & 5 \\ 4 & 2 \end{pmatrix}$.

a. *Calculate $E_1A$, how is your new matrix related to $A$?*

b. *Calculate $E_2A$, how is your new matrix related to $A$?*

c. *Calculate $E_3A$, how is your new matrix related to $A$?*

d. *Calculate $E_5E_4E_2E_3A$, where $E_4 = \begin{pmatrix} 1 & -5 \\ 0 & 1 \end{pmatrix}$ and $E_5 = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix}$, what is special about the matrix that you get?*

e. *Create a vector $b$ with entries equal to the constants in the original system ($2x+5y=7$ and $4x+2y=10$), $b = \begin{pmatrix} 7 \\ 10 \end{pmatrix}$ and calculate $E_5E_4E_2E_3b$. If your original system is $Ax = b$ what is the new system after you perform the above operations? Use this to solve the original system of equations.*

f. *Choose another $b$ and write down the system of equations, what is the solution to this system?*

g. *Let $M = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 0 & 3 \\ 0 & 1 & 0 \end{pmatrix}$. Find elementary row operations and their corresponding elementary matrices such that when $M$ is multiplied on the left by these matrices, the resulting matrix is $I_3$.*

h. *Solve the system $x + 2y = 4,\ 3z = 6,\ y = 8$.*

i. *Let $M = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 3 \\ 0 & 2 & 6 \end{pmatrix}$. What is the reduced row echelon form of $M$? Solve the system $x + 2y = 4,\ y + 3z = 6,\ 2y + 6z = 18$.*

j. *Solve the system $x + 2y = 4, y + 3z = 6, 2y + 6z = 12$ and discuss how your result could be related to the reduced row echelon form of $M$.*

# Lab 3: Powers, Inverses, and Special Matrices

## Introduction

A *square matrix* is an $n \times n$ matrix.

If $A$ is a square matrix and if a matrix $B$ of the same size can be found such that $AB = BA = I$, then $A$ is said to be *invertible* or *nonsingular* and $B$ is called the *inverse* of $A$. If no such matrix $B$ can be found, then $A$ is said to be *singular*.

## Powers of Matrices

Define the matrix $A = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 1 & 0 \\ 0 & 1 & -2 \end{pmatrix}$, $B = \begin{pmatrix} 1 & -1 \\ 9 & 3 \\ 0 & 4 \end{pmatrix}$, $M = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ and $P = \begin{pmatrix} 1 & 4 & 0 \\ 2 & 5 & 0 \\ 3 & 6 & 0 \end{pmatrix}$.

To determine the $k^{th}$ power of a matrix, where $k$ is a positive integer value.

Type: **mpower(The Name of the Matrix,k)** or
**The Name of the Matrix^k**

*Exercises:*

a. *Calculate $A^2$. Is this the same as squaring all the entries in $A$? What is another way to express $A^2$?*

b. *Calculate $B^2$. An error occurred; determine why this error occurred. What property has to hold true in order to take the power of a matrix?*

c. *Determine what matrix $A^0$ is equal to.*

d. *Do the laws of exponents appear to hold for matrices? $A^r A^s = A^{(r+s)}$ and $(A^r)^s = A^{rs}$? Check these by example.*

## Inverse of a Matrix

To determine the inverse of a matrix

Type: **inv(The Name of the Matrix)**

*Exercises:*

a. *Find the inverse of* $A$*,* $A^{-1}$*. What are the dimensions of* $A^{-1}$*? What does* $AA^{-1}$ *equal? What does* $A^{-1}A$ *equal?*

b. *Determine what matrix* $(A^{-1})^{-1}$ *is equal to.*

c. *Calculate* $(AM)^{-1}$*,* $(MA)^{-1}$*,* $A^{-1}M^{-1}$*,* $M^{-1}A^{-1}$*. Which of these matrices are equal?*

d. *Property :* $(A^T)^{-1} = (A^{-1})^T$*. Using the properties you have learned so far, which of the following are equal :* $((AM)^T)^{-1}$*,* $((MA)^T)^{-1}$*,* $(A^{-1})^T(M^{-1})^T$*,* $(M^{-1})^T(A^{-1})^T$*?*

e. *Find the inverse of* $P$*,* $P^{-1}$*. Can you explain why an error occurs? Note that the error is related to the matrix being singular.*

## Special Matrices

A square matrix, $A$, is *symmetric* if $A = A^T$.

A square matrix, $A$, is *diagonal*, if $A_{ij} = 0$ if $i \neq j$.

A square matrix, $A$, is *upper triangular* if $A_{ij} = 0$ when $i > j$ and is *lower triangular* if $A_{ij} = 0$ when $i < j$.

*Exercises:*

a. *Determine what type of matrices* $A + A^T$ *and* $M + M^T$ *are and make a conjecture about a property related to your findings.*

b. *Define* $Q = \begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix}$ *what type of matrix is* $Q^T$*? What type of matrix is* $Q^{-1}$*?*

c. *Find* $Q^2$ *and* $Q^3$*, what type of matrix is* $Q^k$ *for any natural number* $k$*?*

## Theorems and Problems

For each of these statements, either prove that the statement is true or find a counter example that shows it is false.

**Theorem 1**. The inverse of an elementary matrix is an elementary matrix.
**Theorem 2**. If $A$ is invertible then the reduced row echelon form of $A$ is $I$.
**Theorem 3**. If the reduced row echelon form of $A$ is $I$ then $A$ is invertible.
**Theorem 4**. $A$ is a square invertible matrix if and only if $A$ can be written as the product of elementary matrices.
**Problem 5**. If $A$ is invertible then $A^k$ is invertible for any natural number $k$.

**Problem 6**. If $A$ is symmetric so is $A^T$.
**Problem 7**. If $A$ is a symmetric invertible matrix then $A^{-1}$ is symmetric.
**Problem 8**. If $A$ and $B$ are symmetric matrices of the same size then $A+B$ is symmetric.
**Problem 9**. If $A$ and $B$ are symmetric matrices of the same size then $AB$ is symmetric.
**Problem 10**. If $A$ is a square matrix then $A+A^T$ is symmetric.
**Problem 11**. The sum of upper triangular matrices is upper triangular.

# Lab 4: Graph Theory and Adjacency Matrices

## Basics of Graph Theory

A graph consists of vertices and edges. Each edge connects two vertices and we say that these two vertices are *adjacent*. An edge and a vertex on that edge are called *incident*. Given two vertices in a graph $v_1$ and $v_2$, the sequence of edges that are traversed in order to go from vertex $v_1$ to vertex $v_2$ is called a *path* between $v_1$ and $v_2$. Note that there is not necessarily a unique path between vertices in a graph.

A graph can be represented by an *adjacency matrix* where the $ij^{\text{th}}$ entry of the adjacency matrix represents the adjacency between vertex $i$ and vertex $j$. If vertex $i$ and vertex $j$ are adjacent then the $ij^{\text{th}}$ entry is 1, otherwise it is 0.

It is also important to note that there are *directed graphs* and *undirected graphs*. A directed graph's edges are represented by arrows, and the edges of a directed graph can only be traversed in the direction that the arrow is pointing, similar to a one-way street. Here adjacency can also be recognized as being one directional. In an undirected graph, an edge is represented by a line segment and thus adjacency is symmetric.

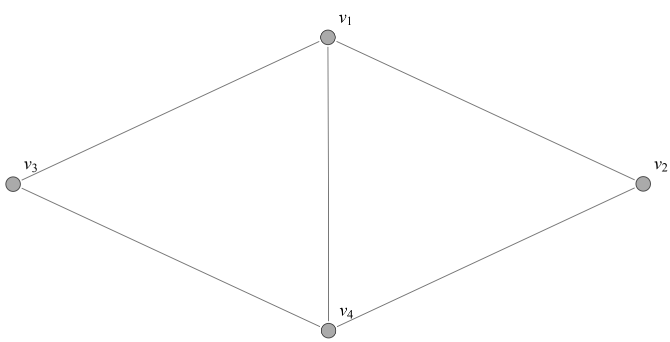


FIGURE 1.1

*Exercises:*

a. *Using the graph in Figure 1.1, create the adjacency matrix, A.*

b. *What type of special matrix is A?*

c. *To create a graph in MATLAB using your adjacency matrix, type :*

***plot(graph(The Name of the Matrix)).***

*Create the graph affiliated with adjacency matrix A using this command.*

d. *How many 1-step paths are there between vertex 1 and vertex 4? How many 2-step paths are there between vertex 1 and vertex 4?*

e. *Calculate* $A^2$ *and discuss how you can determine the number of 2-step paths between vertex 1 and vertex 4 using* $A^2$*.*

f. *The entries of the sum of what matrices would tell you how many paths of 3-steps or less go between vertex 1 and vertex 4?*

## An Application to Hospital Placements in Ghana

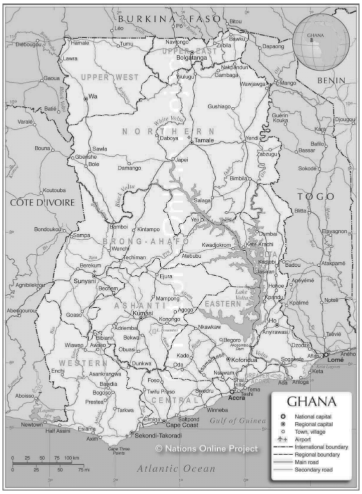


FIGURE 1.2: Map of Ghana

The country of Ghana has national hospitals located in three of its major cities, Accra, Cape Coast, and Techinan. However, many of its citizens from rural villages and small cities can never make it to these city hospitals based on road conditions and other infrastructure issues.

You are a member of the urban health and planning committee for Ghana and would like to strategically place one or two more hospitals in the cities of Dumbai, Damgo, Sunyani, or Kumasi so that all of the villages in the graphical representation of the map in Figure 1.3 can get to a national hospital without passing through more than one additional city. The black cities in Figure 1.3 are the cities in which a proposed hospital can be placed; the gray cities have no hospital and there is no proposal to place one there; and the white cities represent a city that currently has a national hospital.

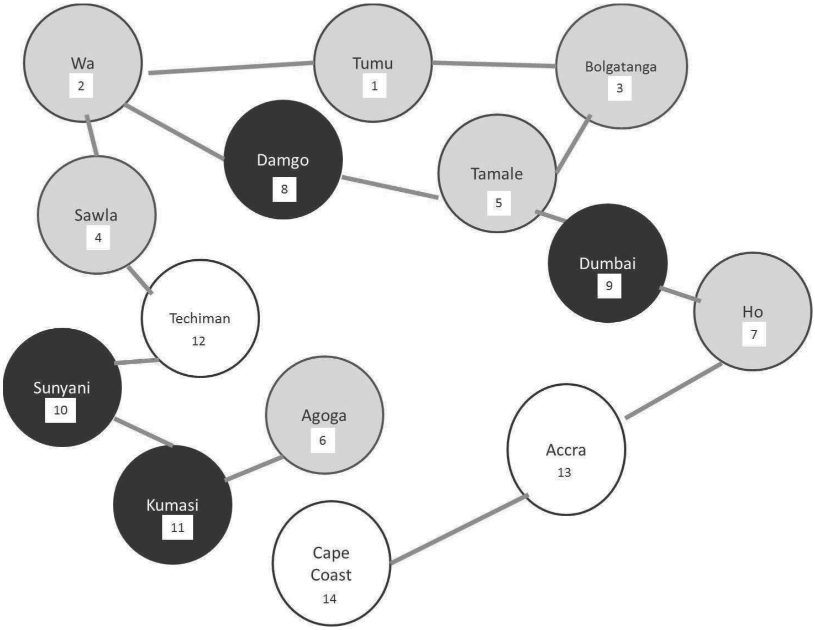


FIGURE 1.3: A graphical representation of the towns

*Exercises:*

a. *Is it currently possible to accomplish the goal of all of the villages on the map, represented by Figure 1.3, having access to a national hospital without passing through more than one additional city? If not what is the maximum number of cities that would have to be traversed in order for the entire population to get to a current hospital? Justify your answer using your knowledge of adjacency matrices and the graph in Figure 1.3.*

b. *What is the minimum number of additional hospitals that can be placed in proposed cities so that people in all of the villages and cities in the graph representation of the map, Figure 1.3, can go to an adjacent city or through at most one other city in order to reach a national hospital? Justify your answer with alterations to your adjacency matrix.*

# Lab 5: Permutations and Determinants

## Permutations

Given a set of elements, $S$, a *permutation* is an ordering of the elements of $S$. The demonstration `http://www.mathworks.com/matlabcentral/fileexchange/64083-permutations-app` shows the permutations as they relate to vertices on a graph. Use this demonstration to answer the following questions.

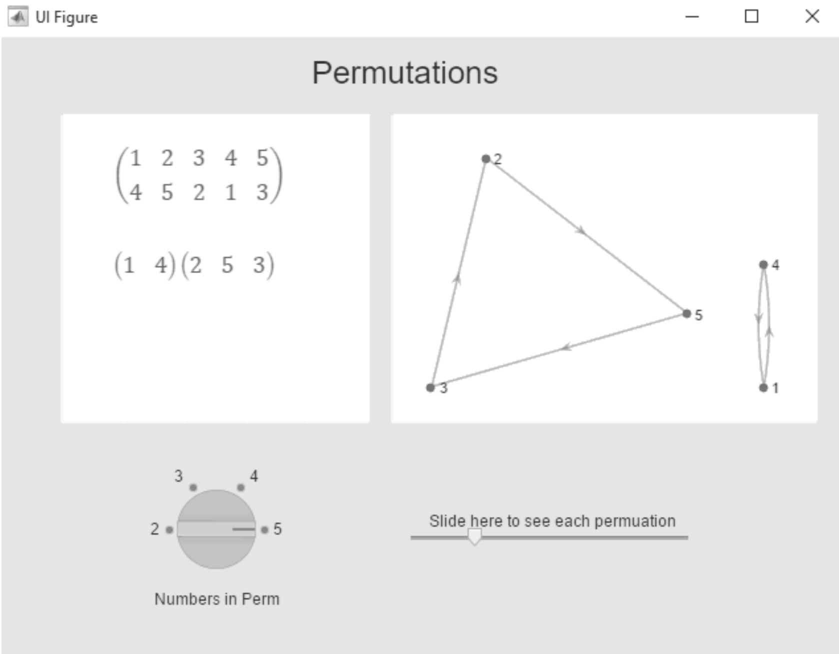


FIGURE 1.4

**Example:** Setting the number length (number of vertices) to 2. There are two notations used to represent the permutations: $\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ and (12). Both of these representations say that the element in the $1^{st}$ position goes to the $2^{nd}$ position and the element in the $2^{nd}$ position goes to position 1. Similarly, $\begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix}$ and (1)(2) leave the elements in the $1^{st}$ and $2^{nd}$ positions.

*Exercises:*

a. *Using the demonstration, write the permutations of 3 elements; how many are there?*

b. *How many permutations of 4 elements do you think there are?*

c. *The sign of a permutation is based on the number of switches that need to be made in order to get the numbers in order. For example, the sign of the permutation* (132) *is* −1 *since we need to make just one switch, of the numbers 2 and 3, to get back to* (123)*. The sign of the permutation* (312) *is 1 since we can get to* (123) *in an even number of steps. Using the demonstration at* `https://www.mathworks.com/matlabcentral/fileexchange/64127-signed-determinant-app` *set the size to 2 and step through the terms (the determinant of a* $2 \times 2$ *is the sum of these terms), discuss how the terms shown here relate to permutations of 2 elements.*

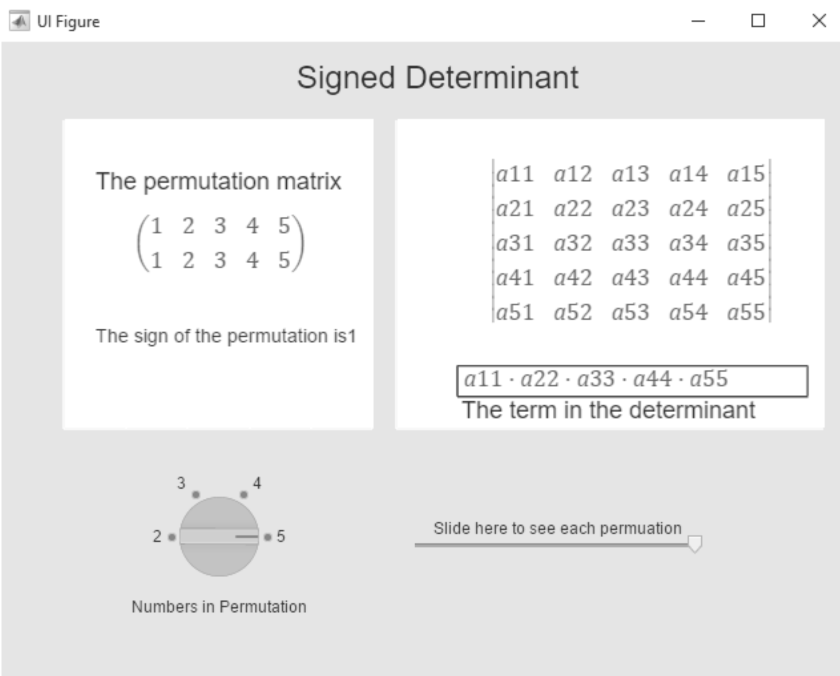


FIGURE 1.5

d. *What do you think the formula for a* $3 \times 3$ *determinant will look like? Use* *your knowledge of permutations on 3 elements to argue your answer and then check your argument with the SignedDeterminant demonstration.*

e. *Change the numbers in* `https://www.mathworks.com/matlabcentral/fileexchange/64140-3x3determinant-app` *to see a trick for doing determinants of* $3 \times 3$ *matrices. Can you state a quick and easy way for doing* $2 \times 2$ *determinants?*

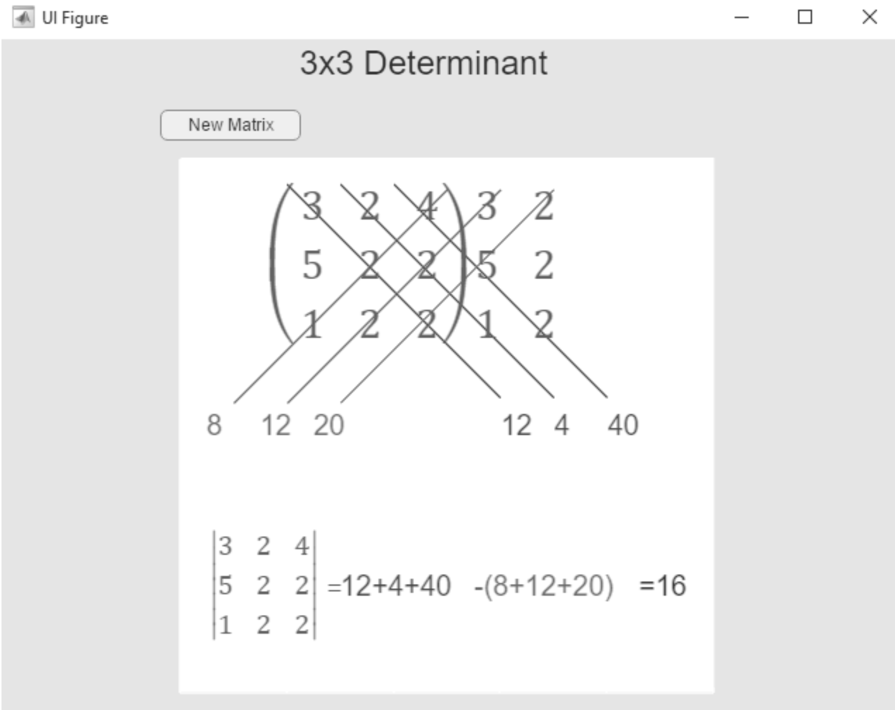


FIGURE 1.6

## Determinants

The *determinant* of a matrix $A$ is denoted $|A|$ or $det(A)$. To calculate

Type: **det(The Name of the Matrix)**

*Exercises: Define* $A = \begin{pmatrix} 2 & 2 & 0 \\ 2 & 1 & 0 \\ 0 & 1 & -2 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 4 \\ 4 & 8 \end{pmatrix}$,
$M = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$, $P = \begin{pmatrix} 1 & 4 \\ 2 & 5 \end{pmatrix}$, $V = \begin{pmatrix} 1 & 0 & 0 \\ 4 & 3 & 0 \\ 0 & 0 & 2 \end{pmatrix}$, *and*
$W = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -1 \end{pmatrix}$.

a. *In Lab 3, we explored the inverse of matrix $A$. Determine the determinant of $A$ and $A^{-1}$ and discuss how they are related.*

b. *Determine the determinant of $B$ and whether or not $B$ is invertible? What do you conjecture about the determinant of matrices that are not invertible?*

c. *Find $det(I_2)$ and $det(I_3)$. Based on these two calculations, what can you conjecture about the value of $det(I_n)$.*

d. *Determine $det(A^T)$ and discuss how this value is related to $det(A)$.*

e. *Determine $det(2A)$, $det(2P)$, $det(3A)$, $det(3P)$ and discuss how they relate to $det(A)$ and $det(P)$.*

f. *We already discovered that matrix multiplication is not commutative, use matrix $A$ and $M$ to decide if $det(AM) = det(MA)$.*

g. *We know that matrix addition is commutative; use matrix $A$ and $M$ to decide if $det(A + M) = det(M + A)$.*

h. *Is $det(A + M) = det(A) + det(M)$?*

i. *Matrix $V$ is a lower triangular matrix and matrix $W$ is a diagonal matrix (and thus also triangular); find the determinants of $V$ and $W$ and discuss how to find determinants of triangular matrices.*

j. *Calculate $\frac{(tr(P))^2 - tr(P^2)}{2}$ and $\frac{(tr(M))^3 - 3tr(M^2)tr(M) + 2tr(M^3)}{6}$ and determine how these quantities relate to $det(P)$ and $det(M)$, respectively.*

The quantities in part j are applications of the Cayley–Hamilton Theorem applied to $2 \times 2$ and $3 \times 3$ matrices.

## Determinants of Elementary Matrices as They Relate to Invertible Matrices

*Exercises: Define* $E_1 = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix}$, $E_2 = \begin{pmatrix} 1 & 0 \\ -4 & 1 \end{pmatrix}$, *and*
$E_3 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$.

a. *If $E_1$ is an elementary matrix representing the operation of multiplying a row by a nonzero scalar, $k = \frac{1}{2}$, find $det(E_1)$. Make a conjecture about how this operation on a matrix effects the determinant of the matrix.*

b. *If $E_2$ is an elementary matrix representing the operation of adding a multiple of a row to another row, find $det(E_2)$. Make a conjecture about how this operation on a matrix effects the determinant of the matrix.*

c. *If $E_3$ is an elementary matrix representing the operation of switching two rows in a matrix, find $det(E_3)$. Make a conjecture about how this operation on a matrix effects the determinant of the matrix.*

## Theorems and Problems

For each of these statements, either prove that the statement is true or find a counter example that shows it is false.

**Thereom 12**. If $det(A)$ is not 0 then $A$ is invertible.
**Theorem 13**. If $A$ is invertible then $det(A)$ is not 0.
**Problem 14**. If $A$ and $B$ are invertible matrices of the same size then $A + B$ is invertible.
**Theorem 15**. If $A$ is a square matrix then $det(A) = det(A^T)$.
**Theorem 16**. If $A$ and $B$ are matrices of the same size then $A$ and $B$ are invertible if and only if $AB$ is invertible.

# Lab 6: $4 \times 4$ Determinants and Beyond

In Lab 5, we discussed how to take the determinant of $2\times2$ and $3\times3$ matrices but what if you have larger matrices for which you have to take the determinant? One technique for finding determinants of larger matrices is called *Cofactor expansion.*

Let's use Cofactor expansion to find the determinant of $A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 2 & 1 & 3 & 1 \\ 0 & 0 & 1 & 4 \end{pmatrix}$.

To Do (Cofactor expansion) :

1. First choose a row or column of your matrix to expand upon. Any row or column will work but as you will see, choosing the row or column with the most 0's is the best choice.
2. Each entry in the matrix has a minor associated with it. The *minor* associated with entry $i$,$j$ is the determinant of the matrix, $M_{ij}$, that is left when the $i^{th}$ row and $j^{th}$ column are eliminated. So for example, $M_{11} = \det\begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 4 \end{pmatrix}$.
3. The determinant of an $n \times n$ matrix, with $ij^{\text{th}}$ entry $a_{ij}$, when expanding about row $i$ is $\sum_{j=1}^{n}(-1)^{(i+j)}a_{ij}M_{ij}$ and when expanding about column $j$ is $\sum_{i=1}^{n}(-1)^{(i+j)}a_{ij}M_{ij}$.

*Exercises:*

a. *Calculate $M_{41}$, $M_{42}$, $M_{43}$, and $M_{44}$ of $A$.*

b. *Use your minors $M_{41}$ through $M_{44}$ to find the determinant of $A$.*

c. *Expand about row 1 to find the determinant of $A$.*

d. *Define* $B = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 3 & 1 \\ 0 & 0 & 1 & 4 \end{pmatrix}$ *and* $P = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & -15 \end{pmatrix}$. *Use cofactor expansion to find $|B|$ and your knowledge of determinants of upper triangular matrices from Lab 5 to find $|P|$.*

e. *Determine elementary matrices $E_1, E_2$, and $E_3$ such that $E_3E_2E_1B = P$.*

f. *In Lab 5, you conjectured about how row operations affect the determinant; use that knowledge along with properties of determinants, and part e., to find* $|B|$.

# Project Set 1

## Project 1: Lights Out

The $5 \times 5$ Lights Out game is a $5 \times 5$ grid of lights where all adjacent lights are connected. Buttons are adjacent if they are directly touching vertically or horizontally (not diagonally). In the Lights Out game, all buttons can be in one of two states, on or off. Pressing any button changes the state of that button and all adjacent buttons. The goal of this project is to create a matrix representation of the Lights Out game where all lights start on and need to be turned off. A picture of the Lights Out game with buttons labeled can be found in Table 1.1.

**TABLE 1.1**
5x5 Lights Out Grid

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 6 | 7 | 8 | 9 | 10 |
| 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 | 25 |

a. Note that since in the Lights Out game a button changes its own state when pressed, a button is adjacent to itself. Create the adjacency matrix, $M$, for the $5 \times 5$ game in Table 1.1.

b. A *row vector* is a $1 \times n$ matrix and a *column vector* is an $n \times 1$ matrix. If $i$ is the initial state vector, what would the column vector $i$ look like? Recall the goal is to determine if all lights can be turned off, starting with all lights on. (Use 0 for off and 1 for on.)

c. If $f$ is the final state vector, determine $f$.

d. Write up your findings and supporting mathematical argument.

## Project 2: Traveling Salesman Problem

Joe's Pizzeria wishes to send a single driver out from its main store who will make 4 deliveries and return to the store at the end of the route.

a. A *weighted adjacency matrix* is an adjacency matrix whose entries represent the weights of the edges between two adjacent vertices. For example, the weights in Figure 1.7 represent the time it takes to travel from one site, vertex, to another site. Create a weighted adjacency matrix, $A$, with the Joe's Pizzeria as vertex 1. $A_{ij}$ should represent the time traveled by the driver between site $i$ and site $j$.

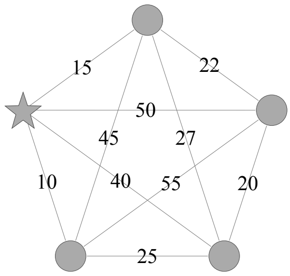


FIGURE 1.7: Map of delivery sites and Joe's Pizzeria denoted by a star

b. As mentioned before, the driver should start and end at the pizzeria while stopping at each of the delivery sites. The time of one such path is $A_{12} + A_{23} + A_{34} + A_{45} + A_{51}$. Calculate the time that the driver travels if the driver travels on this path. This path is using the off diagonal of $A$.

c. Other paths can easily be explored by looking at permutations of the rows of the matrix $A$. How many permutations are there?

d. The command **idx=perms([5 4 3 2 1])** will create a list of all permutations of the numbers 1 through 5, and the loop

```
I=eye(5);
for c = 1:120
P=I(idx(c,:),:)
disp(P*A*transpose(P))
end
```

should produce all of the matrices which are permutations of the rows, and respective columns, of $A$.

If **P=I(idx(2,:),:)** and **B=P*A*transpose(P)**, use the off diagonal of $B$ to determine another route that the driver can take and the time that the truck takes to traverse this route.

e. Write a small for loop utilizing the commands from part d. to find the path that gives the quickest route. Write up your findings and supporting mathematical argument.

## Project 3: Paths in Nim

The demonstration found at `https://www.mathworks.com/matlabcentral/fileexchange/64175-counting-paths-of-nim-app` shows the number of paths (limited to a length of $r$) between point $A$ in row 1 and $B$ in row $r$ in the game of Nim with $n$ rows. Your problem is to determine a matrix representation to determine the number of paths shown in this demonstration.

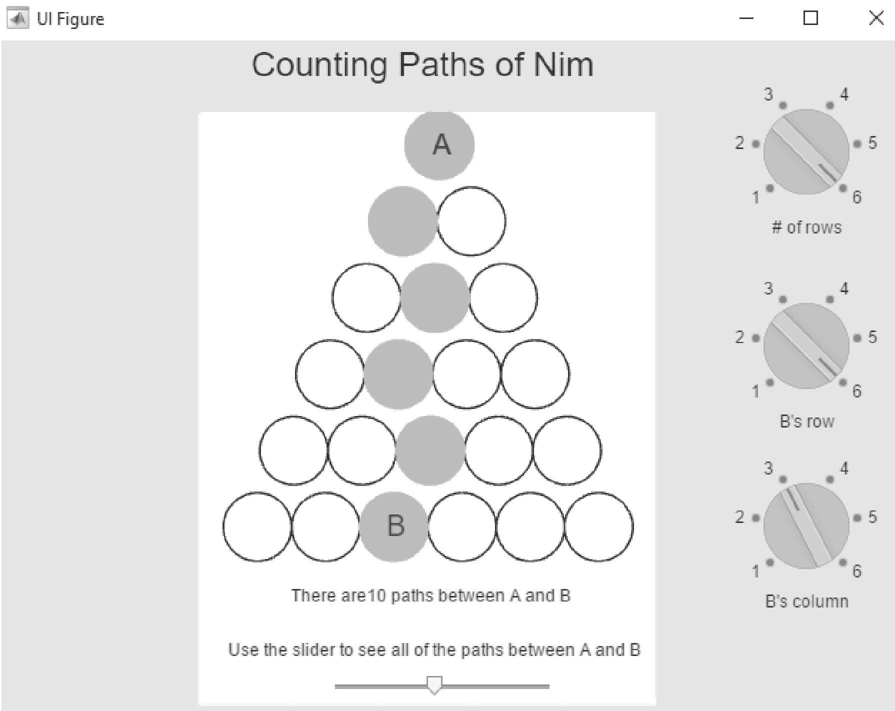


FIGURE 1.8: The Nim Board

a. If you did not care how long the path is from point $A$ to point $B$ (that is, the length is not limited by the number of rows, $r$), determine a matrix representation to count the number of 2-step paths, 3-step paths, and k-step paths. For simplicity allow $n$, the total number of rows in Nim, to be fixed at 5.

b. Make a conjecture about the number of k-step paths between a point $A$ in row 1 and point $B$ when $B$ is position $(row,column) = (r,c)$ when there are 5 rows and in general $n$ rows in the Nim game.

c. Using what you found, create a representation limiting the length of the path between $A$ and $B$, as in the demonstration.

## Project 4: Gaussian Elimination of a Square Matrix

Project 4 requires some programming in *MATLAB*. A small sample program is provided below which retrieves a matrix, $A$, and divides the first row by $a_{11}$.

$prompt =' Input\ a\ matrix\ A';$
$A = input(prompt)$
$n = size(A);$
$temp = A(1,1);$
$for\ j = 1 : n(2)$
$A(1,j) = A(1,j)/temp;$
$end$
$disp(A)$

a. Create a program (assuming that rows need not be swapped for Gaussian Elimination– that is assume no 0's will show up on the main diagonal) to get any square matrix $A$ in row echelon form. Since we are only doing Gaussian Elimination of square matrices here, you may want to include an if-then statement that checks that the matrix is square.

b. Create a program where swaps are allowed to get any square matrix $A$ in row echelon form.

c. Create a program where swaps are allowed to get any square matrix $A$ in reduced row echelon form.

## Project 5: Sports Ranking

In the 2013 season, the Big Ten football games in Table 1.2 occurred with W representing the winner. The question is how to rank these teams based on these games. The *dominance matrix*, $A$, is a matrix of zeros and ones where $A_{i,j} = 1$ if teams $i$ and $j$ played and team $i$ won and $A_{i,j} = 0$ otherwise.

a. Create the dominance matrix and determine all one-step dominances for each team and one- and two-step dominances for each team combined.

b. Rank-order the teams by number of victories and by dominance.

c. Consider the dominance rankings of Minnesota and Michigan State. How is it possible that Minnesota has a higher dominance ranking than Michigan State while Minnesota has fewer victories than Michigan State?

**TABLE 1.2**
2013 Big Ten Results

| | |
|---|---|
| Michigan State W – Indiana | Michigan State W – Purdue |
| Michigan State W – Illinois | Michigan State W – Iowa |
| Indiana W – Penn State | Penn State W – Michigan |
| Iowa W – Minnesota | Iowa W – Northwestern |
| Michigan W – Minnesota | Michigan W – Indiana |
| Minnesota W – Northwestern | Minnesota W – Wisconsin |
| Minnesota W – Nebraska | Nebraska W – Purdue |
| Nebraska W – Illinois | Ohio State W– Wisconsin |
| Ohio State W – Penn State | Ohio State W – Iowa |
| Ohio State W – Northwestern | Wisconsin W – Illinois |
| Wisconsin W – Northwestern | Wisconsin W – Purdue |

d. Given that many times in a league every team does not necessarily play every other team, would ranking victories or dominance seem more reasonable for national rankings? How might one incorporate the score of the game into the dominance ranking as well?

## Project 6: Archaeological Similarities, Applying Seriation

In archeology, *seriation* is a relative dating method in which assemblages or artifacts from numerous sites, in the same culture, are placed in chronological order. Most data that is collected is binary in nature where if an artifact, or record, possesses an identified trait, the artifact would be assigned a one for that trait and a zero otherwise.

In this project, there are 4 artifacts and 5 traits: Artifact A has Traits 1, 2, and 4. Artifact B has Traits 1, 3, 4, and 5, Artifact C has Traits 1, 2, 3, and 4, and Artifact D has Traits 1, 4, and 5.

a. Create a binary matrix, $M$, with rows representing artifacts and columns representing traits that the artifacts may possess.

b. $S = MM^T$ is called the *similarity matrix.* Find the similarity matrix and describe what $S_{i,i}$ and $S_{i,j}$ where $i \neq j$ represent.

c. $D = N - S$ where $N$ is a matrix with all entries equal to $n$, where $n$ is the number of traits. $D$ is called the *dissimilarity matrix.* Many researchers who use seriation techniques attempt to find an ordering that minimizes some cost function. One cost function of interest is the number of dissimilarities. The dissimilarity between artifact $i$ and artifact $j$ is $D_{i,j}$, so the dissimilarity cost of an ordering of $m$ artifacts $1,2,3,\ldots,m$ is $D(1,2,3,\ldots,m) = D_{1,2} + D_{2,3} + D_{3,4} + \cdots + D_{m-1,m} + D_{m,1}$. Find the dissimilarity matrix using matrix $M$ and the dissimilarity cost for the ordering of artifacts $\{A,B,C,D\}$.

d. Find the dissimilarity cost for the ordering of artifacts $\{A,C,B,D\}$. How many unique orderings of these artifacts are there? Explore these different orderings and determine the ordering that minimizes the dissimilarity cost.

## Project 7: Edge-Magic Graphs

A graph is called *edge-magic* if the edges can be labeled with positive integer weights such that (i) different edges have distinct weights, and (ii) the sum of the weights of edges incident to each vertex is the same; this sum is called the *magic constant.*

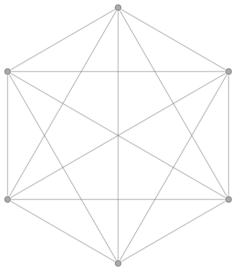

FIGURE 1.9

a. For the graph in Figure 1.9, create a system of linear equations that would determine the edge weights if the magic constant is 40.

b. Use your system from part a. to determine a solution, distinct edge weights of value 1 through 15, that produce a magic constant of 40. Recall that all edge weights must be nonzero.

c. The graph in Figure 1.9 is called the *complete graph* with 6 vertices, denoted $K_6$. In a complete graph with $n$ vertices, denoted $K_n$, each pair of vertices is adjacent. Make a conjecture about edge-magic properties of $K_n$.

# *MATLAB* Demonstrations and References

## *MATLAB* Demonstrations by Crista Arangala

All of the following MATLAB demonstrations are posted on the MATLAB Community File Exchange.

1. *Matrix Multiplication App*, `https://www.mathworks.com/matlabcentral/fileexchange/63993-matrix-multiplication-app`

2. *Permutations App*, `http://www.mathworks.com/matlabcentral/fileexchange/64083-permutations-app`

3. *Signed Determinant App*, `https://www.mathworks.com/matlabcentral/fileexchange/64127-signed-determinant-app`

4. $3 \times 3$ *Determinant App*, `https://www.mathworks.com/matlabcentral/fileexchange/64140-3x3determinant-app`

5. *Counting Paths of Nim App*, `https://www.mathworks.com/matlabcentral/fileexchange/64175-counting-paths-of-nim-app`

6. *Inverse and Nullspaces in Gf(p)*, `https://www.mathworks.com/matlabcentral/fileexchange/65139-inverse-and-nullspaces-in-gf-p`

7. *Hill Cipher App*,`https://www.mathworks.com/matlabcentral/fileexchange/63769-hill-cipher-app`

8. *Transforming the Dog*, `https://www.mathworks.com/matlabcentral/fileexchange/64916-transforming-the-dog`

9. *Transforming the Dog with Rotation*, `https://www.mathworks.com/matlabcentral/fileexchange/64917-transforming-the-dog-with-rotation`

10. *Transforming the Dog with a Composition of Linear Transformations*, `https://www.mathworks.com/matlabcentral/fileexchange/66107-transforming-the-dog-with-a-composition-of-linear-transformations`

11. *Sum of Two Vectors*, https://www.mathworks.com/matlabcentral/fileexchange/64926-sum-of-two-vectors

12. *Triangle Inequality with Functions*, https://www.mathworks.com/matlabcentral/fileexchange/64935-triangle-inequality-with-functions

13. *Cauchy–Schwarz for Vectors*, https://www.mathworks.com/matlabcentral/fileexchange/64939-cauchy-schwarz-for-vectors

14. *Cauchy–Schwarz for Integrals*, https://www.mathworks.com/matlabcentral/fileexchange/64954-cauchy-schwarz-inequality-for-integrals

15. *Change of Basis*, https://www.mathworks.com/matlabcentral/fileexchange/64955-change-of-basis

16. *Least Squares Linear Regression*, https://www.mathworks.com/matlabcentral/fileexchange/64960-least-square-linear-regression

17. *Conic Sections*, https://www.mathworks.com/matlabcentral/fileexchange/64976-conic-sections

18. *Multi-state Lights Out*, https://www.mathworks.com/matlabcentral/fileexchange/65109-multistate-lights-out

19. *Orthogonal Grids*, https://www.mathworks.com/matlabcentral/fileexchange/65197-orthogonal-grids

20. *Singular Values*, https://www.mathworks.com/matlabcentral/fileexchange/65264-singular-values

21. *Homogeneous Systems of Coupled Linear Differential Equations*, https://www.mathworks.com/matlabcentral/fileexchange/64494-homogeneous-systems-of-coupled-linear-differential-equations

22. *Visualizing the Solution of Two Linear Differential Equations*, https://www.mathworks.com/matlabcentral/fileexchange/64580-visualizing-the-solutions-of-two-linear-differential-equations

23. *Predator-Prey Model*, `https://www.mathworks.com/matlabcentral/fileexchange/64676-predator-prey-system`

24. *Forced Oscillator with Damping*, `www.mathworks.com/matlabcentral/fileexchange/64747-forced-oscillator-with-dampening`

25. *A Simple Epidemic Model*, `https://www.mathworks.com/matlabcentral/fileexchange/64768-simple-epidemic-model`

## References


1. [C. Arangala et al. 2014], J. T. Lee and C. Borden, "Seriation algorithms for determining the evolution of The Star Husband Tale," *Involve*, 7:1 (2014), pp. 1-14.

2. [C. Arangala et al. 2010], J. T. Lee and B. Yoho, "Turning Lights Out," *UMAP/ILAP/BioMath Modules 2010: Tools for Teaching*, edited by Paul J. Campbell. Bedford, MA: COMAP, Inc., pp. 1-26.

3. [Atkins et al. 1999], J. E. Atkins, E. G. Boman, and B. Hendrickson, "A spectral algorithm for seriation and the consecutive ones problem," *SIAM J. Comput.* 28:1 (1999), pp. 297-310.

4. [D. Austin, 2013], "We recommend a singular value decomposition," A Feature Article by AMS, http://www.ams.org/samplings/feature-column/fcarc-svd, viewed December 12, 2013.

5. [N.T.J. Bailey, 1950], "A simple stochastic epidemic," *Biometrika*, Vol. 37, No. 3/4, pp. 193-202.

6. [E. Brigham, 1988], *Fast Fourier Transform and Its Applications*, Prentice Hall, Upper Saddle River, NJ, 1988.

7. [G. Cai and J. Huang, 2007], "A new finance chaotic attractor," *International Journal of Nonlinear Science*, Vol. 3, No. 3, pp. 213-220.

8. [P. Cameron], *"The Encyclopedia of Design Theory,"* http://www.designtheory.org/library/encyc/topics/had.pdf, viewed December 17, 2013.

9. [D. Cardona and B. Tuckfield, 2011], "The Jordan Canonical Form for a class of zero-one matrices," *Linear Algebra and Its Applications*, Vol. 235 (11), pp. 2942-2954.

10. [International Monetary Fund], *World Economic Outlook Database*, http://www.imf.org/external/pubs/ft/weo/2013/01/weodata/index.aspx, viewed December 20, 2013.

11. [J. Gao and J. Zhung, 2005], "Clustering SVD strategies in latent semantics indexing," *Information Processing and Management* 21, pp. 1051-1063.

12. [J. Gentle, 1998], *Numerical Linear Algebra with Applications in Statistics*, Springer, New York, NY, 1998.

13. [L. P. Gilbert and A. M. Johnson, 1980], "An application of the Jordan Canonical Form to the Epidemic Problem," *Journal of Applied Probability*, Vol. 17, No. 2, pp. 313-323.

14. [D. Halperin, 1994], "Musical chronology by Seriation," *Computers and the Humanities*, Vol. 28, No. 1, pp. 13-18.

15. [A. Hedayat and W. D. Wallis, 1978], "Hadamard matrices and their applications," *The Annals of Statistics*, Vol. 6, No. 6, pp. 1184-1238.

16. [K. Bryan and T. Leise, 2006], The "$25,000,000,000 Eigenvector," in the education section of *SIAM Review*, August 2006.

17. [J. P. Keener, 1993], "The Perron-Frobenius Theorem and the ranking of football teams," *SIAM Review*, Vol. 35, No. 1. (Mar., 1993), pp. 80-93.

18. The Love Affair of Romeo and Juliet, http://www.math.ualberta.ca/ ∼ devries/crystal/ContinuousRJ /introduction.html, viewed December 22, 2013.

19. [I. Marritz, 2013] "Can Dunkin' Donuts really turn its palm oil green?," *NPR*, March 2013, viewed December 11, 2013. `http://www.npr.org/blogs/thesalt/2013/03/12/174140241/can-dunkin-donuts-really-turn-its-palm-oil-green`.

20. [P. Oliver and C. Shakiban, 2006], *Applied Linear Algebra*, Prentice Hall, Upper Saddle River, NJ, 2006.

21. [One World Nations Online], Map of Ghana, http://www.nationsonline.org/oneworld/map/ghana_map.htm, viewed December 10, 2013.

22. [Rainforest Action Network], "Truth and consequences: Palm oil plantations push unique orangutan population to brink of extinction," http://www.npr.org/blogs/thesalt/2013/03/12/174140241/can-dunkin-donuts-really-turn-its-palm-oil-green, viewed December 11, 2013.

23. [K. R. Rao and P. C. Yip, 2001], *The Transform and Data Compression Handbook*, CRC Press, Boca Raton, FL, 2001.

24. [L. Shiau, 2006], "An application of vector space theory in data transmission," *The SIGCSE Bulletin.* 38. No 2, pp. 33-36.

25. [A. Shuchat, 1984], "Matrix and network models in archaeology," *Mathematics Magazine*. 57. No 1, pp. 3-14.

26. The University of North Carolina Chemistry Department, *Balancing Equations Using Matrices*, http://www.learnnc.org/lp/editions/chemistry-algebra/7032, viewed December 9, 2013.

27. Figure 6.3, http://www.scholarpedia.org/article/File:Equilibrium_figure_summary_2d.gif