

# **Importing & Exporting Data**

## **Low Level Functions**

## **GUI & App Designer**

**LAB 8**

**ENGG100 - Spring 2024**



# Lab 8 Objectives

- Be able to read, write and manipulate data from files
- Use low-level functions to manipulate data
- Use MATLAB's App Designer to solve problems

# Importing & Exporting Data

- Spreadsheets:

Command	Purpose
<b>xlsinfo</b>	Determine if file contains MS Excel spreadsheet
<b>xlsread</b>	Read MS Excel spreadsheets
<b>xlswrite</b>	Write to MS Excel spreadsheet

- Comma-seperated values (CSV):

Command	Purpose
<b>csvread</b>	Read csv file
<b>csvwrite</b>	Write to csv file
<b>readmatrix</b>	Create an array by reading column-oriented data from a file
<b>dlmwrite</b>	Write matrix to ASCII-delimited file
<b>textscan</b>	Read formatted data from text file or string
<b>type</b>	Display contents of the file

# TASK 1 - SCRIPT

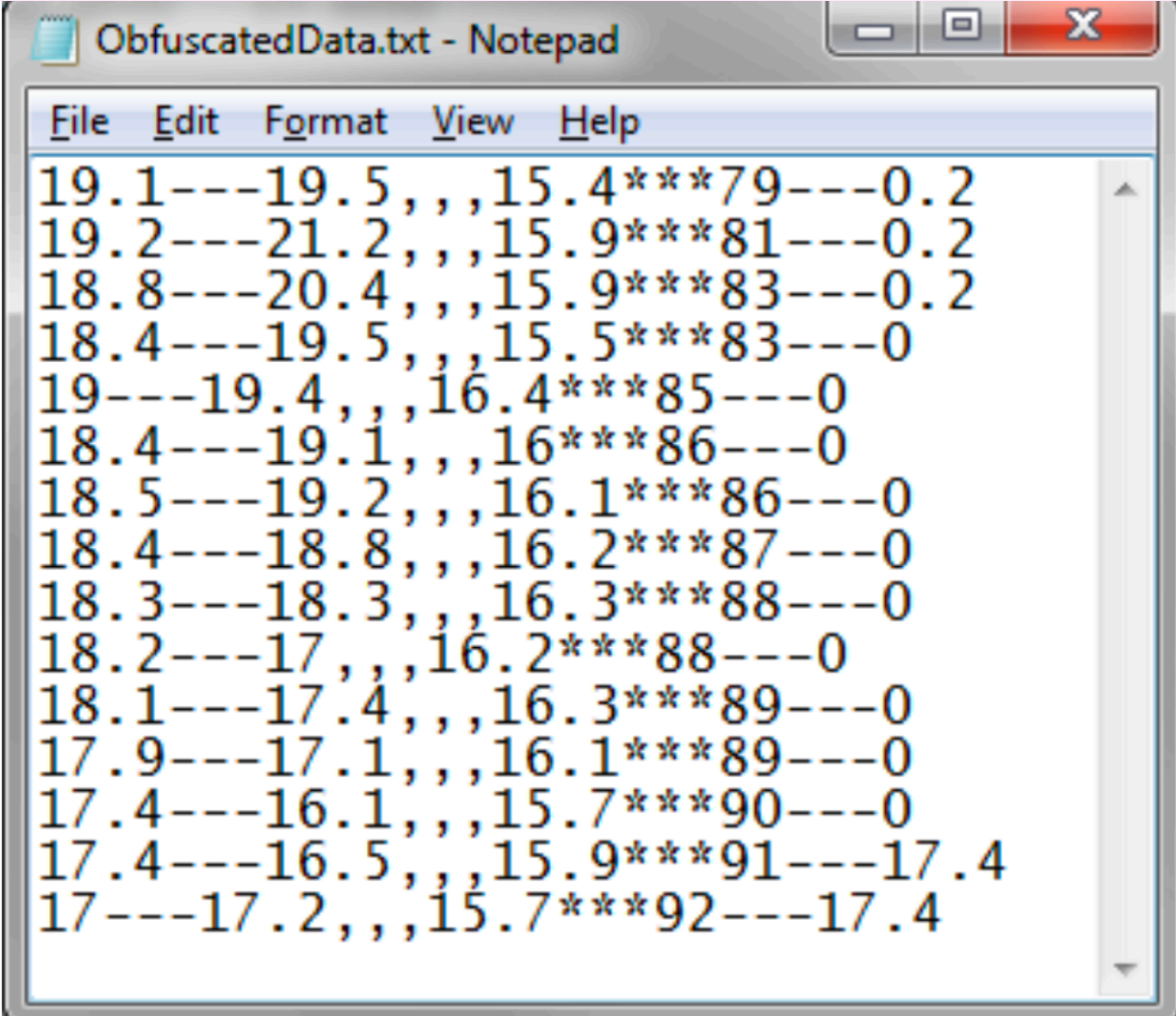
- Create a magic square of size 20x20 and using the **xlswrite** function, export the data from the matrix to an excel spreadsheet (.xlsx)
- Use the **csvwrite** function to export the data to a comma separated variable file (.csv)
- Then use the **dlmwrite** function to export the data to a text file (.txt) using the "pipe" delimiter | (vertical bar), instead of a comma
- **For your report** - attach all 3 files OR screenshots (excel, csv, text), along with the code to your submission



# Low Level I/O Functions

- Sometimes, we might not have standard format for files, and we may need to read the file character by character

Command	Purpose
<b>fclose</b>	<b>Close one or all open files</b>
<b>feof</b>	<b>Test for end-of-file</b>
<b>ferror</b>	Information about file I/O errors
<b>fgetl</b>	Read line from file, removing newline characters
<b>fgets</b>	Read line from file, keeping newline characters
<b>fileread</b>	Read contents of file into string
<b>fopen</b>	<b>Open file, or obtain information about open files</b>
<b>fprintf</b>	Write data to text file
<b>fread</b>	Read data from binary file
<b>frewind</b>	Move file position indicator to beginning of open file
<b>fscanf</b>	<b>Read data from text file</b>
<b>fseek</b>	Move to specified position in file
<b>ftell</b>	Position in open file
<b>fwrite</b>	Write data to binary file



A screenshot of a Notepad window titled "ObfuscatedData.txt". The window displays a list of obfuscated data entries, each consisting of a numerical value followed by a series of dashes, commas, and asterisks, and ending with a numerical value. The entries are as follows:

```
19.1---19.5,,,15.4***79---0.2
19.2---21.2,,,15.9***81---0.2
18.8---20.4,,,15.9***83---0.2
18.4---19.5,,,15.5***83---0
19---19.4,,,16.4***85---0
18.4---19.1,,,16***86---0
18.5---19.2,,,16.1***86---0
18.4---18.8,,,16.2***87---0
18.3---18.3,,,16.3***88---0
18.2---17,,,16.2***88---0
18.1---17.4,,,16.3***89---0
17.9---17.1,,,16.1***89---0
17.4---16.1,,,15.7***90---0
17.4---16.5,,,15.9***91---17.4
17---17.2,,,15.7***92---17.4
```

# TASK 2 - SCRIPT

- For Task 2, you will need to output the same 20x20 matrix created in Task 1 to a text file, but you will need to use low-level File I/O functions so that you can incorporate three plus signs between each value
- You can read through the low-level file I/O documentation for functions such as fopen, fclose and fprintf using MATLAB docs/help
- For your report: Show your code and the text file that contains your magic square

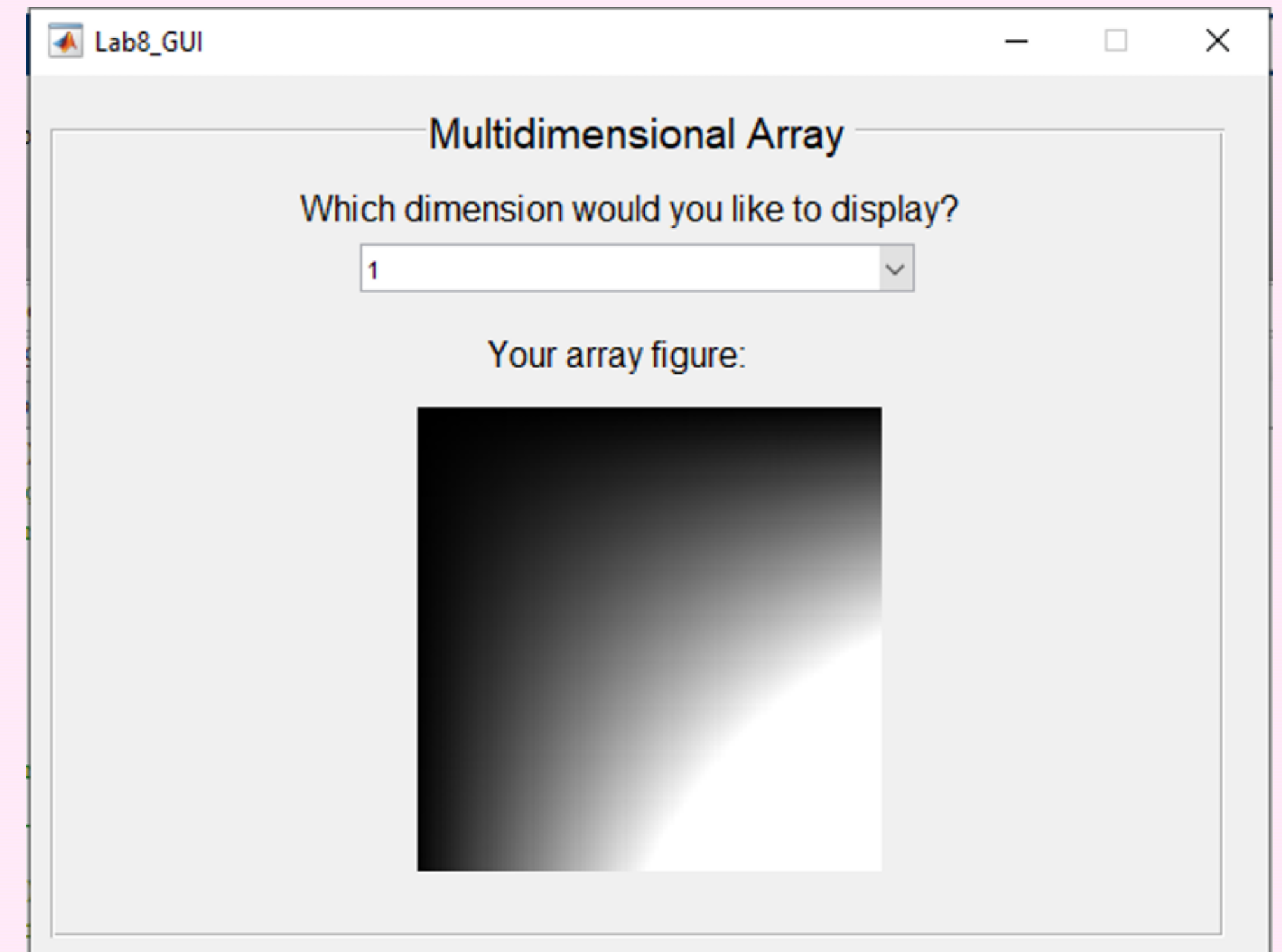
**Magic Square**

**400+++21+++41+++340+++320+++101+++121+++260+++240+++181+++20**



# TASK 3 - SCRIPT

- Create a 100x100x100 numerical 3D multidimensional array where each value in each array corresponds to the multiplication of its indices multiplied by 0.001 (use nested for loops)
- Create a MATLAB GUI that asks the user from a pop-up menu, which of the three dimensions they would like to display. This refers to the third axes of the array (user can choose from 1,2 or 3; referring to `Array(:,:,X)`, where X is the chosen number)
- Display the array in an axes as a grayscale image - example of the GUI shown here —>
- **For your report** - show a screenshot of your code view & all 3 figures for 1,2 and 3



# TASK 4 - SCRIPT

- Create a GUI that uses a function created for a projectile launched from a canon above
- The user should input various variables for  $x_0$ ,  $y_0$ ,  $v_{0x}$  and  $v_{0y}$ , then the user clicks on "redraw plot", and the figure should update in the GUI
- Your GUI should use the following components:
  - Edit Text
  - Static Text
  - Axes
  - Push Button
- An example of what your GUI should look like is shown —>

