Last (family) name:_____

First (given) name:_____

I have neither given nor received aid on this examination, nor concealed any violation of the Honor Code.
The only online resources I have used during this exam are those listed below as well as the following sites (list URL and exam problem #):

Signature: _____

ID Number: _____

EECS 551 Midterm 1, 2020-09-29 5PM EDT (24 hour online)

- There are 8 problems for a total of 56 points.
- This part of the exam has 3 pages. Make sure your copy is complete.
- This is a 24-hour "online" exam. Many of the exam questions will be on Canvas; this pdf has additional questions that you should answer by submitting to gradescope unless instructed otherwise in each problem.
- This is an "open book" exam. During the exam, you may use all of the course materials on the Canvas EECS 551 site including the course notes on google drive, as well as wikipedia, the built-in JULIA help, and the online JULIA manual. If you use resources from any other web site then you must cite the source along with your honor code statement above.
- You may use without rederiving any of the results presented in the course notes.
- You must complete the exam entirely on your own.
- If you need an exam question to be clarified, post a private question to the instructors on piazza.
- Clearly box your final answers. For full credit, show your complete work clearly and legibly. Answers must be submitted properly to gradescope to earn credit.
- For multiple-choice questions, select *all* correct answers.
- To "disprove" any statement, provide a concrete counter-example. For maximum credit, make that counter-example as small and simple as possible, *e.g.*, having the smallest possible matrix dimensions and using the simplest numbers like "0" and "1" as much as possible. For example, to disprove the statement "any square matrix $\boldsymbol{A}$ is invertible," the smallest and simplest counter-example is the 1×1 matrix $\boldsymbol{A} = [0]$.

[6] **1**. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be vectors in $\mathbb{F}^9$ with Euclidean norms 5 and 2, respectively, and define the matrices $\boldsymbol{A} = \boldsymbol{x}\boldsymbol{y}'$ and $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{A}'$. Either determine $\mathsf{trace}\{\boldsymbol{B}\}$, or explain why more information is needed to determine it.

[6] **2**. Write a JULIA expression that evaluates $\det\left\{ \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{D} \end{bmatrix}' \begin{bmatrix} \boldsymbol{E} & \boldsymbol{F} \\ \boldsymbol{0} & \boldsymbol{H} \end{bmatrix} \right\}$ as efficiently as possible.
(Assume that $\boldsymbol{A}$,$\boldsymbol{D}$,$\boldsymbol{E}$,$\boldsymbol{H}$ are square, and that $\boldsymbol{B}$ and $\boldsymbol{F}$ and the two $\boldsymbol{0}$ matrices are **conformable**.)

[6] **3**. Prove or disprove. Every square rank-1 matrix is diagonalizable.

[6] **4**. Let $\boldsymbol{A} \in \mathbb{F}^{N\times N}$ be invertible with SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}'$ and let $\boldsymbol{x} \in \mathbb{F}^K$ denote a vector with $\|\boldsymbol{x}\|_2 = 7$. Express $\left\|\boldsymbol{A}^{-1}\boldsymbol{u}_1\boldsymbol{x}'\right\|_{\mathrm{F}}$ concisely in terms of the given ingredients, or explain why more information would be needed for a concise expression.

[6] **5**. Let $\boldsymbol{A}$ be a diagonalizable matrix with eigenvalues $-2, 3+4\imath$. Determine σ_1 or explain why more information is needed to determine it.

[6] **6**. The unit ball $\mathcal{B}_1$ in a vector space $\mathcal{V}$ is the set of vectors having norm less than unity: $\mathcal{B}_1 = \{\boldsymbol{x} \in \mathcal{V} \,:\, \|\boldsymbol{x}\|_2 \leq 1\}$. Determine the span of the unit ball in $\mathbb{F}^N$, *i.e.*, $\mathsf{span}(\mathcal{B}_1)$, or explain why more information is needed to determine it.

[6] **7**. Let $\boldsymbol{A} = \boldsymbol{x}\boldsymbol{y}'$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are two nonzero vectors. Express the orthogonal complement of the nullspace of $\boldsymbol{A}$ in terms of the vectors $\boldsymbol{x}$ and/or $\boldsymbol{y}$.

[14] **8**. Some image processing operations require discrete approximations to *second* derivatives. For unit sample spacing, in 1D the most popular approach is

$$\ddot{f}(x) \approx 2f(x) - f(x-1) - f(x+1).$$

For a 1D signal sampled at $x_1, \ldots, x_N$, with $x_n - x_{n-1} = 1$, we can express this relation for all x_i samples via a matrix-vector product with a $N \times N$ second finite-difference matrix $\boldsymbol{S}_N$ as follows:

$$\begin{bmatrix} \ddot{f}(x_1) \\ \ddot{f}(x_2) \\ \vdots \\ \ddot{f}(x_N) \end{bmatrix} \approx \boldsymbol{S}_N \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \end{bmatrix}, \qquad \boldsymbol{S}_N \triangleq \begin{bmatrix} -1 & 2 & -1 & & & & \\ -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 2 & -1 \\ & & & & -1 & 2 & -1 \end{bmatrix}.$$

Note that the first and last rows use a slightly different approximation due to the lack of a $f(0)$ or $f(N+1)$ sample. In this problem you will derive and implement the analog of $\boldsymbol{S}_N$ for 2D differentiation. Let $f(x, y)$ be a function of

two variables. We can approximate its 2nd partial derivatives using finite differences as follows:

$$\frac{\partial^2}{\partial x^2} f(x,y) \approx 2f(x,y) - f(x+1,y) - f(x-1,y) \tag{1}$$

$$\frac{\partial^2}{\partial y^2} f(x,y) \approx 2f(x,y) - f(x,y+1) - f(x,y-1). \tag{2}$$

To simplify notation, define the $m \times n$ matrices `FXY`, `SFDX`, and `SFDY` having elements as follows:

$$\texttt{FXY}[m,n] = f(m,n), \qquad \texttt{SFDX}[m,n] = \frac{\partial^2}{\partial x^2} f(m,n), \qquad \texttt{SFDY}[m,n] = \frac{\partial^2}{\partial y^2} f(m,n), \qquad \begin{matrix} m = 1,\ldots,M \\ n = 1,\ldots,N. \end{matrix}$$

The x coordinate is along the column of `FXY` and the y coordinate is along the row of `FXY`, so we can think `FXY[x,y]`. Define corresponding vectors `fxy`, `sfdx`, and `sfdy` in $\mathbb{R}^{MN}$ to be vectorized versionsof `FXY`, `SFDX`, and `SFDY`. With this notation, we succinctly express (1) and (2) in terms of an appropriately sized matrix $\boldsymbol{A}$ as:

$$\begin{bmatrix} \texttt{sfdx} \\ \texttt{sfdy} \end{bmatrix} = \boldsymbol{A}\ \texttt{fxy}.$$

- Write a function `second_diff_2d` that takes as input the dimensions M and N of $M \times N$ array `FXY` and returns the appropriate $\boldsymbol{A}$ matrix. Your code should work even when M and N are quite large. To save memory, the elements of $\boldsymbol{A}$ should be of type `Int8`. Here is a docstring and template for your code.

```
"""
   A = second_diff_2d(M, N)

In:
- `M` and `N` are positive integers

Out:
- `A` is a appropriate matrix such that `A * X[:]` computes the
second finite differences down the columns (along x direction)
and across the (along y direction) of the `M x N` matrix `X`,
where `eltype(A) == Int8`
"""
function second_diff_2d(M, N)
```

- [9 points] Upload your code for the function `second_diff_2d` as a single, complete `uniquename.jl` file to Canvas under the "Midterm1code" assignment. (Do not upload any test or plotting code to Canvas.)
- [5 points] You should test your code yourself by writing and running your own test routine *before* uploading your solution. Submit to gradescope a screenshot of your test code, along with some plot(s) or image(s) that provide evidence that your code was working before you uploaded it to Canvas.

end