

MAT 116E

PROJECT-4

Projects must be submitted on the ninova system until 21th december before 13:00 pm.

What to turn in: Copy the text from your scripts and paste it into a document. If a question asks you to plot or display something to the screen, also include the plot and screen output your code generates. Submit either a *.doc or *.pdf file.

Write a simple shuffling 'encryption' algorithm.

- a. Open a new script and save it as `encrypt.m`
- b. At the top of the script, define the *original* string to be: This is my top secret message!
- c. Next, let's shuffle the indices of the letters. To do this, we need to make a string of encoding indices
 - i. Make a vector that has the indices from 1 to the length of the original string in a randomly permuted order. Use **randperm** and **length**
 - ii. Encode the original string by using your encoding vector as indices into *original*. Name the encoded message *encoded*.
- d. Now, we need to figure out the decoding key to match the encoding key we just made.
 - i. Assemble a temporary matrix where the first column is the encoding vector you made in the previous part and the second column are the integers from 1 to the length of the original string in order. Use **length**, and you may need to transpose some vectors to make them columns using **'**.
 - ii. Next, we want to sort the rows of this temporary matrix according to the values in the first column. Use **sortrows**.
 - iii. After it's been sorted, extract the second column of the temporary matrix. This is your decoding vector.
 - iv. To make the *decoded* message, use the decoding vector as indices into *encoded*.
- e. Display the original, encoded, and decoded messages
 - i. Display the following three strings, where : *original*, *encoded*, and *decoded* are the strings you made above. Use **disp**
Original: *original*
Encoded: *encoded*
Decoded: *decoded*
- f. Compare the original and decoded strings to make sure they're identical and display the result
 - i. Use **strcmp** to compare the *original* and *decoded* strings. Name the output of this operation *correct*. *correct* will have the value 1 if the strings match and the value 0 if they don't
 - ii. Display the following string: Decoded correctly (1 true, 0 false): *correct* use **disp** and **num2str**
- g. Run the script a few times to verify that it works well. You should see an output like this:

```
>> encrypt
Original: This is my top secret message!
Encoded : sisrpigct tessaoheem m yTse!
Decoded : This is my top secret message!
Decoded correctly (1 true, 0 false): 1
```