

You have 2 hours to complete this test.

There are 4 multi-part questions in this test. Create a script named `test2.m` to answer the questions 2–4. For each part of each question, create a new cell (section) in your script. You do not need to publish a report from your script.

You have access to the internet during this test. Please do not use any communication software such as email, text chat, social media services, etc.

When you are finished your test, submit your functions and script using the online submit service <https://webapp.eecs.yorku.ca/submit/>

1. (10 points) A point  $(x, y)$  is below, on, or above of the line  $x = y$  under the following conditions:

below  $x > y$   
on  $x - y = 0$   
above  $x < y$

- (a) (6 points) Create and complete the following MATLAB function (you can cut and paste the code into a new function):

```
function belowabove = classify(x, y, tol)
%CLASSIFY Below, on, or above the line x = y.
% BELOWABOVE = CLASSIFY(X, Y, TOL) classifies a scalar point (X, Y)
% as being below, on, or above of the line x = y.
% BELOWABOVE is -1 if the point is below the line,
% 0 if the point is on the line, and 1 if the point is above
% the line.
%
% A point is considered on the line if (X - Y) is inside the
% range -TOL to +TOL.
```

- (b) (4 points) Create a new function `vclassify` that is similar to the function in part (a) except that `x` and `y` are now vectors of x-coordinates and y-coordinates, respectively. For example,

```
>> vclassify([3, 1, -1], [0, 1, 5], 1e-6)
ans =
    -1     0     1
```

This can be done without a loop, although you may use loops if you want. You should include suitable help comments.

2. (14 points) Create a script named `test2.m` to answer questions 2–4.

(a) Consider the following MATLAB code:

```
result = zeros(1, length(vec));
for idx = 1:length(vec)
    if vec(idx) < 0
        result(idx) = 1;
    end
end
```

- i. (2 points) In a comment, explain using 1–2 sentences what the code is trying to accomplish.
- ii. (2 points) Rewrite the code using a single MATLAB statement.

(b) Consider the following MATLAB code:

```
result = 0;
[r, c] = size(A);
n = r * c;
for row = 1:r
    for col = 1:c
        result = result + A(row, col);
    end
end
result = result / n;
```

- i. (2 points) In a comment, explain using 1–2 sentences what the code is trying to accomplish.
- ii. (2 points) Rewrite the code using a single MATLAB statement.

(c) (6 points) Using nested loops (like in part (b)) write a script that computes the same matrix as `diag(v)` for some vector `v`. You should not assume anything about the length of `v` except that the length is greater than zero. You should not use any MATLAB functions (especially the function `diag`). Note that `diag(v)` makes a square matrix of all zeros except for the main diagonal which contains the elements of `v`; for example:

```
>> diag([1, 2, 3, 4, 5])
ans =

     1     0     0     0     0
     0     2     0     0     0
     0     0     3     0     0
     0     0     0     4     0
     0     0     0     0     5
```

3. (14 points) Congratulations! You've gotten a summer job working as a research assistant in a professor's lab. Your first task is to analyze some data collected by a graduate student in the lab. The data is stored in a file as a table having two columns; an example of a good data file is shown below:

measurement of X	measurement of Y
10.23	126.3
9.833	138.43
10.7514	120.531

(and so on)

Unfortunately, the equipment used in the experiments experienced some intermittent failures and some (much less than half) of the measurements have been recorded with the incorrect values; whenever an X measurement is incorrect, the corresponding Y measurement is also incorrect, and vice versa. Your job is to find the incorrect measurements, set the incorrect measurements to NaN, and then perform a simple analysis of the data.

To answer this question, create a script named `test2.m` and complete the following parts:

- (4 points) Save a copy of the file <http://www.eecs.yorku.ca/course/1541/test2/baddata.txt>. Read the contents of the file using `textread`.
- (2 points) Create a histogram of all of the measurements; you do not need to label the plot. Based on the histogram results, how can you determine the correct X and Y measurements? Record your answer in your script.
- (4 points) Based on your answer to part (b), find the incorrect X and Y measurements, and set them to NaN. Record the steps in your script so that I can repeat your answer in MATLAB.
- (2 points) Create a boxplot of the corrected measurements. Label the x axis using the labels 'X' and 'Y' for the two groups of measurements. You do not need a label on the y axis or a title. If you are having difficulty with part (b) then create a boxplot of the original measurements.
- (2 points) Remove the NaN values from data; you can remove NaN values from a vector `v` using the following statement:

```
v(isnan(v)) = [];
```

Compute the mean and standard deviation of the corrected measurements after removing the NaN values.

4. (12 points)

- (a) (3 points) Consider the following logical statement for a scalar  $x$ :  $\sim(x < 0 \mid x > 10)$   
Rewrite the statement using an AND statement (hint: consider the mathematical interpretation of the statement).
- (b) (3 points) There are two types of loops in MATLAB: **while** loops and **for** loops. What is the rule for choosing which type of loop you should use?
- (c) (3 points) Suppose that you create a boxplot of a single set of measurements; what does the boxplot tell you about the accuracy *and* precision of the measurements? Assume that you know the true value of the thing that is being measured.
- (d) (3 points) Consider the following user-defined function:

```
function [s] = csum(x)
%CSUM Cumulative sum using a loop.

s = x(1);
for idx = 2:length(x)
    spre = s(idx - 1);
    s = [s (spre + x(idx))];
end

end
```

The function appears to be correct, and gives the same result as the MATLAB function `cumsum`, but it is also much slower than `cumsum`, especially when  $x$  becomes very long. Explain why the function is slow, and suggest a simple change to the code that will speed up the loop (you must still use a loop).