

---

```

% Pierce Zhang, CMOR220, FALL 2023, Competency matrices and matrix
% creation
% matrices_creation.m
% Answers to matrices and matrix creation competency
% Last modified: 2 September 2023

% Driver to answer the questions
function matrices_creation
    % Problem 1
    V_1_a = even_space_vectors(10, 15, 75); disp(length(V_1_a));
    V_1_b = even_space_vectors(1, 20, 1500); disp(length(V_1_b));
    V_1_c = even_space_vectors(0.5, 1, 50); disp(length(V_1_c));

    % Problem 2
    V_2_a = even_space_vectors_trad(5, 0.5, 15); disp(length(V_2_a));
    V_2_b = even_space_vectors_trad(1, 0.1, 1500); disp(length(V_2_b));
    V_2_c = even_space_vectors_trad(-500, 5, 500); disp(length(V_2_c));

    % Problem 3
    A = [11:16];
    A = [A ; 21:26];
    A = [A ; 31:36];
    A = [A ; 41:46] % original A
    Amod = prob3_modify(A)

    % Problem 4
    B = [(1:7)'];
    B = [B (11:17)'];
    B = [B (21:27)'];
    B = [B (31:37)'] % original B
    Bmod = prob4_modify(B)

    % Problem 5
    prob5()
end

% Inputs: a, starting value; b, ending value; n, desired number of entries
% Outputs: v, vector from a to b with n entries evenly-spaced
function [v] = even_space_vectors(a, b, n)
    % Uses built-in linspace command to generate a vector v of
    % evenly-distributed numbers with a as starting value in first cell and
    % b as ending value in last cell.
    v = linspace(a, b, n);
end

% Inputs: a, starting value, int, interval, b, ending value
% Outputs: v, vector from a to b with int, interval between a and b
function [v] = even_space_vectors_trad(a, int, b)
    % Uses built-in vector constructor to generate a vector v of numbers
    % with int separating them starting with a in the first cell and b in
    % the final cell.
    v = a:int:b;

```

---

---

```

end

% Inputs: A, matrix ideally the one specified in driver function as 'A'
% Outputs: Amod, modified matrix according to problem 3 specs
function [Amod] = prob3_modify(A)
    Amod = A;
    Amod(1,1) = 0; Amod(1,6) = 0; Amod(4,1) = 0; Amod(4,6) = 0;
    Amod(2, :) = [];
    Amod(:, 3) = [];
    Amod(2, 3) = 100;
end

% Inputs: B, matrix ideally the one specified in driver function as 'B'
% Outputs: Bmod, modified matrix according to problem 4 specs
function [Bmod] = prob4_modify(B)
    Bmod = B;
    Bmod(1:2:end,1) = 100;
    Bmod(2:2:end,2) = 200;
    Bmod(1:2:end,3) = 300;
    Bmod(1:end,4) = 400;
end

% Inputs: none
% Outputs: none
function prob5()
    % This function has no output; however, it will display three values to
    % the console.
    % Based on vector V and M as specified in problem 5, it will create a
    % row vector of zeros of same length as V, a column vector of same
    % length as V, and a ones matrix of same size as M.
    V = [1 1.5 2 2.5 3 3.5 4 4.5 5];
    M = rand([3,5]);

    zeros_rowV = zeros(1,length(V))
    col_zeros_V = zeros(length(V),1)
    ones_M = ones(size(M))
end

```

75

1500

50

21

14991

201

A =

11	12	13	14	15	16
----	----	----	----	----	----

---

21	22	23	24	25	26
31	32	33	34	35	36
41	42	43	44	45	46

$A_{mod} =$

0	12	14	15	0
31	32	100	35	36
0	42	44	45	0

$B =$

1	11	21	31
2	12	22	32
3	13	23	33
4	14	24	34
5	15	25	35
6	16	26	36
7	17	27	37

$B_{mod} =$

100	11	300	400
2	200	22	400
100	13	300	400
4	200	24	400
100	15	300	400
6	200	26	400
100	17	300	400

$zeros\_rowV =$

0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---

$col\_zeros\_V =$

0
0
0
0
0
0
0
0
0
0

$ones\_M =$

---

<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>
<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>
<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>	<i>1</i>

*Published with MATLAB® R2023a*