

Mock exam exercises

1. Write a program that asks the user to introduce the coordinates of three points. Next, the program asks the user to select two of the three points and computes their distance.

Example of an execution:

Introduce X coordinate of point 1: 0

Introduce Y coordinate of point 1: 0

Introduce X coordinate of point 2: 3

Introduce Y coordinate of point 2: 3

Introduce X coordinate of point 3: 2

Introduce Y coordinate of point 3: 2

Introduce the number of the first point: 1

Introduce the number of the second point: 2

The distance between points 1 and 2 is 4.24

Solution:

```
clear;
points = zeros(3,2);
points(1,1) = input('Introduce X coordinate of point 1: ');
points(1,2) = input('Introduce Y coordinate of point 1: ');
points(2,1) = input('Introduce X coordinate of point 2: ');
points(2,2) = input('Introduce Y coordinate of point 2: ');
points(3,1) = input('Introduce X coordinate of point 3: ');
points(3,2) = input('Introduce Y coordinate of point 3: ');
p1 = input('Introduce the number of the first point: ');
p2 = input('Introduce the number of the second point: ');
distance = sqrt((points(p1,1) - points(p2,1))^2 + (points(p1,2) -
points(p2,2))^2);
fprintf('\n The distance between points %d and %d is %.2f', p1, p2,
distance);
```

- Write a program that asks for three numbers. Next, it asks for two numbers between 1 and 3. Finally, it asks for a number between 1 and 2. The program should print the last number introduced on screen, the number in the second set of numbers that corresponds to that number following the order of introduction, and the number in the first set of numbers that corresponds to the number of the second set just printed.

Example of an execution:

First set of numbers

Introduce a number: 12

Introduce a number: 56

Introduce a number: 9

Second set of numbers

Introduce a number between 1 and 3: 2

Introduce a number between 1 and 3: 3

Now introduce a number between 1 and 2: 2

You have introduced the number 2, which corresponds to the value 3 in the second set of numbers, and the value 9 in the first set of numbers.

Solution:

```
clear;
disp('First set of numbers');
v1 = zeros(1,3);
v1(1) = input('Introduce a number: ');
v1(2) = input('Introduce a number: ');
v1(3) = input('Introduce a number: ');
disp('Second set of numbers');
v2 = zeros(1,2);
v2(1) = input('Introduce a number between 1 and 3: ');
v2(2) = input('Introduce a number between 1 and 3: ');
number = input('Now introduce a number between 1 and 2: ');
fprintf('\n You have introduced the number %d, which corresponds to the value
%d in the second set of numbers, and the value %d in the first set of
numbers', number, v2(number), v1(v2(number)));
```

3. Write a program that asks the user to introduce 4 characters, and stores them in a 2 x 2 matrix. Then, it asks the user to introduce the row number of the matrix that he/she wants to display. Finally, the program prints the characters of the corresponding row on screen.

Example of an execution:

First character: a
Second character: b
Third character: c
Fourth character: d
Which row do you want to display? 2
On row 2 you find the characters c and d.

Solution:

```
clear;
c(1) = input('First character: ', 's');
c(2) = input('Second character: ', 's');
c(3) = input('Third character: ', 's');
c(4) = input('Fourth character: ', 's');
c = [c(1) c(2); c(3) c(4)];
row = input('Which row do you want to display? ');
fprintf('On row %d you find the characters %c and %c.\n', row, c(row,1),
c(row,2));
```

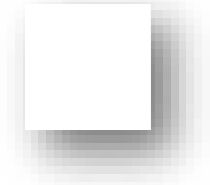
4. You are given a 20 x 20 pixels B/W image, i.e. the values of the pixels can be either 0 (pixel is black/B) or 1 (pixel is white/W). This image can be loaded into your program as a matrix as follows: **IMG = imread("SAMPLE_BW_IMAGE.bmp")**. You are required to create a program that performs the following operations:
 - a. Erase the whole image. Note that erasing is equivalent to turning all pixels to 1.
 - b. Create a square whose corners are (2,2) (18,2) (2,18) (18,18).
 - c. Erase the content of the elements of the first vertical half of the image created in step b (first ten pixels on the left).
 - d. Create something that looks like an "H" letter.

Note: You can test the results by showing the image using **imshow(IMG)**

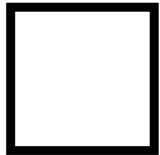
Example of an execution:



Next operation? Y/N:



Next operation? Y/N:



Next operation? Y/N:



Next operation? Y/N:



Solution:

```
clear;
IMG = imread("SAMPLE_BW_IMAGE.bmp");
imshow(IMG);
% a) Erase the whole image
IMG(:) = 1;
imshow(IMG);
DONEXT = input('NEXT OPERATION? Y/N: ', 's');
% b) Create a square
IMG(2,2:18) = 0;
IMG(18,2:18) = 0;
imshow(IMG);
IMG(2:18,2) = 0;
IMG(2:18,18) = 0;
```

```
imshow(IMG);
DONEXT = input('NEXT OPERATION? Y/N: ', 's');
% c)Erase half of the image created in b)
IMG(:,1:10) = 1;
imshow(IMG);
DONEXT = input('NEXT OPERATION? Y/N: ', 's');
% d) Create something that looks like an H
IMG(:) = 1 ;
IMG(4:16,4) = 0;
IMG(4:16,16) = 0;
IMG(10,4:16) = 0;
imshow(IMG);
```