Programming Skills Mock Exam Questions

Dr Jon Shiach

These mock exam questions have been designed to be similar to the question that you will receive in the exam. In the exam you will have a choice to use MATLAB or Python to answer the questions but it is advisable to become proficient in both languages for future units and to enhance your employability.

Questions

1. (a) Given the matrices

$$A = \begin{pmatrix} 1 & 1 & 2 \\ 3 & 3 & 1 \\ 1 & 7 & 4 \end{pmatrix}, \qquad B = \begin{pmatrix} 5 & 6 & 7 \end{pmatrix}$$

write a program that does the following:

(i) defines arrays corresponding to A and B and print the arrays;

(ii) replaces the second row of A with the elements from B and prints the result;

(iii) print the elements from the final column of the matrix from part (ii).

(b) Define a function called `inv_2x2` that calculates the inverse of a 2×2 matrix using the formula

$$A^{-1} = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}.$$

Your program should use an `if` statement to check if A is singular, i.e., $a_{11}a_{22} - a_{12}a_{21} = 0$, and print a warning message if so. Use your function to calculate the inverse of the matrix

$$C = \begin{pmatrix} 1 & -1 \\ 3 & 2 \end{pmatrix}.$$

2. (a) Define arrays corresponding to the following matrices:

$$A = \begin{pmatrix} 3 & 0 \\ -1 & 5 \end{pmatrix}, \qquad B = \begin{pmatrix} 2 & 0 & -1 \\ 4 & 2 & 3 \end{pmatrix}.$$

Hence calculate and output:

(i) AB (the matrix mulitplication of A and B);

(ii) $(AB)^T A$;

(iii) $\det(A)$;

(iv) A^{-1};

(v) the matrix formed using the first two columns of B which is then multiplied by A.

(b) Write a program that uses nested `for` loops to calculate the sum of the elements of the matrix $C = B^T B$.

(c) The determinant of a 2×2 matrix A is defined by the following

$$\det(A) = a_{11}a_{22} - a_{12}a_{21}.$$

Write a function called `mydet` that given an input of a 2×2 array `A` returns its determinant. Your function should check whether the inputted matrix has 2 rows and 2 columns, if it does not not it should print a warning message and exit. Test your function on your arrays `A` and `B` from part (a).

3. (a) Write a program to calculate the sum of the following series using a given number of terms

$$e^{2x} = \sum_{n=0}^{\infty} \frac{(2x)^n}{n!} = 1 + 2x + \frac{(2x)^2}{2!} + \frac{(2x)^3}{4!} + \dots$$

and output the result correct to 6 decimal places. Your program may use the `factorial` command from MATLAB or the `math` library in Python. Use your program to calculate e^{2x} for $x = 2.5$ using the first 12 terms.

(b) Write a program that uses a `while` loop to calculate the sum of the series in part (a) ceasing iterations when the difference between successive terms of the series is less than 10^{-4} and outputs the value of e^{2x} and the number of iterations used. Test your program using $x = 2.5$.

4. A sequence of numbers is generated using the following scheme

$$x_{n+1} = \frac{x_n^2 - 1}{2x_n - 3},$$

for some given starting value x_0. For example, if $x_0 = 2$ then the sequence is

$$\begin{aligned} x_0 &= 2, \\ x_1 &= 3, \\ x_2 &= 2.6667, \\ x_3 &= 2.6190, \\ &\vdots \end{aligned}$$

(a) Write a program that outputs the sequence up to x_8 in the form of a table with a column containing the values of n and another containing the value of x_n correct to 6 decimal places, i.e.,

```
 n       xn
--------------
 0    x.xxxxxx
 1    x.xxxxxx
 :       :
 2    x.xxxxxx
```

Check your program using $x_0 = 2$.

(b) Write a program that it uses a `while` loop to calculate the sequence from part (a) stopping when the difference between two consecutive values is less than 10^{-6}.

(c) Define a function called `mysequence` that given inputs of the starting value x_0 and the number of terms in the sequence n returns an array containing the values in the sequence, for example if $x_0 = 2$ and the number of terms is 4 then the array should be `[ 2, 3, 2.6667, 2.6190 ]`. Use your function to calculate the first 12 numbers in the sequence with a starting value of $x_0 = 100$.

5. The following programs draws a red circle of radius 10 centred at the point $(15, 25)$.

MATLAB:

```
x = zeros(37, 1);
y = zeros(37, 1);

for i = 1 : 181
    x(i) = 10 * cos(pi * i) - 15;
    y(i) = 10 * sin(pi * i) + 25;
end

plot(x, y, 'r')

axis([ -20, 20, -20, 20 ])
axis square
xlabel('x')
ylabel('y')
```

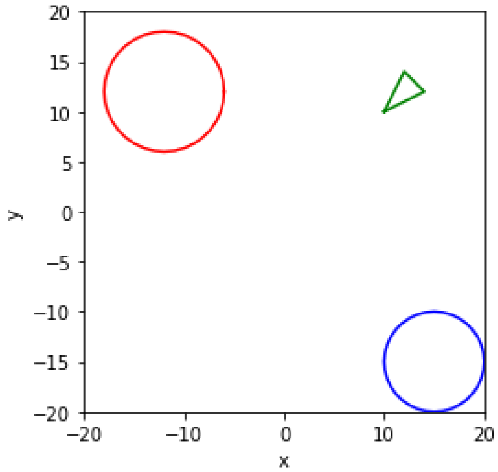
Python:

```
import numpy as np
import matplotlib.pyplot as plt

fig, ax = plt.subplots()

x = np.zeros(37)
y = np.zeros(37)
for i in range(37):
    x[i] = 10 * np.cos(np.pi / 18 * i) + 4
    y[i] = 10 * np.sin(np.pi / 18 * i) + 2

ax.plot(x, y, 'r')

ax.set_xlim(-20, 20)
ax.set_ylim(-20, 20)
ax.set_aspect('equal')
ax.set_xlabel('x')
ax.set_ylabel('y')

plt.show()
```

(a) Amend this program to draw the following objects on the **same** axes

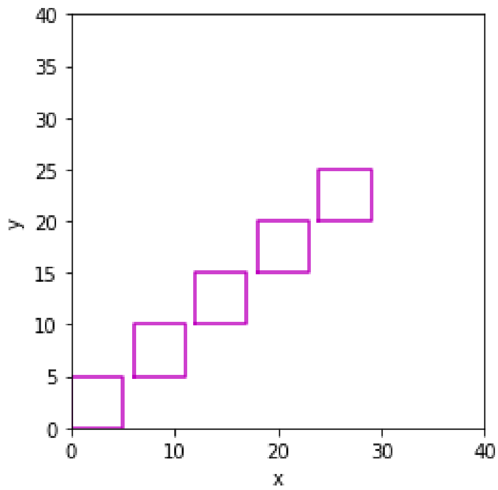
- a red circle with radius 6 and centre $(-12, 12)$;
- a blue circle with radius 5 and centre $(15, -15)$;
- a green triangle with corners at $(10, 10)$, $(14, 12)$ and $(12, 14)$.

Your plot should look like the following:



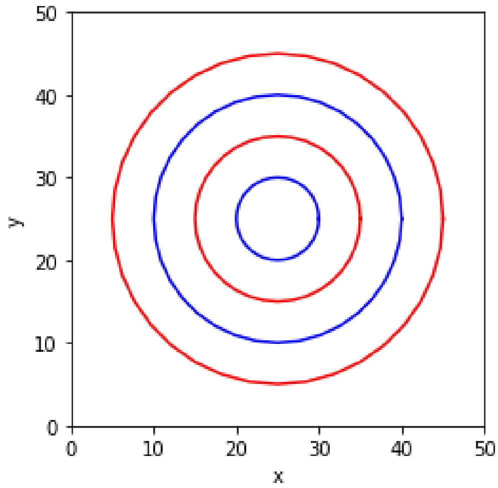
(b) Write a program that plots a set of 5 magenta squares, each of side length 2 units on a set of axes where $x \in [0, 40]$. The bottom left corner of the first square should be positioned at the origin. The other 4 squares should be positioned **above and to the right** of the first square with a **horizontal gap of 1 unit** and a **vertical gap of 0 units** . You **must use a loop** to draw the 5 squares rather than set of repeated statements.

Your plot should look like the following:



(c) Write a program to draw set of 5 circles, each with centre at $(25, 25)$ on a set of axes where $x, y \in [0, 50]$. The circles should decrease in radius from 20 to 5 in steps of -5. The circles should appear in alternate colours, red and blue, with the largest circle drawn in red. Again you **must use a loop** to draw the 5 circles and use an if statement to draw the circles in the correct colour.

Your plot should look like the following:


50
40
30
20
10
0
y
0
10
20
30
40
50
x

6. (a) The following expression defines a sequence of x values

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{5}{x_n}\right).$$

For example, given a starting value of $x_0 = 1$

$$\begin{aligned} x_0 &= 1, \\ x_1 &= 3, \\ x_2 &= 2.3333, \\ x_3 &= 2.2381, \\ &\vdots \end{aligned}$$

Write a program to calculate the values in this sequence using a starting value of $x_0 = 5$ upto x_5 and stores them in a one-dimensional array. Use a print command to output the array.

(b) Using a **for loop**, produce a table of this sequence with a column for the the values of n and another column for the values of x_n correct to 7 decimal places, i.e.,

```
 n     xn
--------------
 0  x.xxxxxxx
 1  x.xxxxxxx
 :      :
 5  x.xxxxxxx
```

(c) Define a function called `mysequence2` that calculates the following sequence given inputs for the starting value x_0, the value of b and the maximum value of n, and stores the values in a one-dimensional array.

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{b}{x_n}\right).$$

Using your function, produce **four** tables of this sequence up to x_4 for $x_0 = b$ and the value of b changing from 7 to 13 in steps of 2. You **must use a for loop** to loop through the values of b.

7. (a) Write a program that defines a two-dimensional array which is equivalent to the following matrix. Using a pair of nested 'for' loops calculate and output the number of 1s in the matrix.

$$X = \begin{pmatrix} 1 & 0 & 1 \\ 2 & -3 & 2 \\ 1 & 0 & -2 \end{pmatrix}$$

Your output should appear in the following form

```
number of 1s = xx
```

(b) Define a function called `distance` that given inputs of the co-ordinates of two points in $\mathbb{R}^2$ as two-element arrays of the form (x_1, y_1) and (x_2, y_2), calculates and returns the distance between the two points using the formula

$$\text{distance} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Use your function to calculate the distance between the points $(1.3, 5.7)$ and $(7.9, -0.8)$.

(c) The area of a triangle defined in $\mathbb{R}^2$ with vertices (x_1, y_1), (x_2, y_2) and (x_3, y_3) can be calculated using the formula

$$\text{area} = \frac{1}{2} \det \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix}.$$

Define a function called `triangle_area` that given inputs to the vertex co-ordinates calculates the area of the triangle. You may use the `det` command from MATLAB or the NumPy library in Python. Use your function to calculate the area of the triangle with vertices $(0, 1)$, $(5, 2)$ and $(2, 7)$.

8. (a) The following expression gives the probability of correctly choosing 6 lottery balls from a possible 59,

$$\frac{1}{59} \times \frac{2}{58} \times \frac{3}{57} \times \frac{4}{56} \times \frac{5}{55} \times \frac{6}{54}.$$

Write a program which calculates and outputs this value to **10 decimal places** with suitable preceding text on the same line. The program **must use a for loop** to perform the calculation.

(b) The series below calculates the value of $\cos(x)$ for a given value of x

$$\cos(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots$$

Write a program to calculate the value of this series given the x value and number of terms. You may use the `factorial` command from MATLAB or the `math` library in Python. Use your program to calculate the value of $\cos(0.5)$ using 4 terms and print the result correct to **5 decimal places**.

(c) Define a function called `mycos` that uses the series from part (b) and a **while loop** to calculate the value of $\cos(x)$ ceasing iterations when the **absolute** difference between successive approximations is less than 10^{-6}. Use your function to calculate $\cos(1)$ and output the result correct to **8 decimal places**.