

MATH 2221 **B** Mathematics Laboratory II

Lab Assignment 8

Name: _____

Student ID.: _____

In this assignment, you are asked to run **MATLAB** demos to see **MATLAB** at work. The color version of this assignment can be found in your own **H:**\ drive.

Instructions

1. Start **MATLAB**, until you see a window with the **MATLAB** prompt "**>>**". This window is called the **Command Window**.
2. After you started have **MATLAB**, you will automatically be in the directory **H:**\. Please enter "**diary on**" after the **MATLAB** prompt **>>** only once to record all your work in **H:**\diary. No marks will be given if no diary is found.
3. Enter "**demo**" after the prompt **>>**. You will see a new window with many things to play with. This is the **Demo Window**.
4. In the **Demo Window**, try to locate figures or problems similar to those in the exercises below. Then locate the commands that generate these figures or problems. Try them in the **Command Window**. Just enter (or cut and paste) the commands after **>>** to see what happens.
5. You should write your results on the lab sheet provided, and save the figures in the **H:** drive, in your personal drive.
6. Please read and sign the following declaration before handing in your assignment. Otherwise, no marks will be given.

I declare that the assignment here submitted is original except for source material explicitly acknowledged. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained on the website <http://www.cuhk.edu.hk/policy/academichonesty/>

Signature

Date

1 (20 marks)		3 (25 marks)	
2 (25 marks)		4 (30 marks)	
		Total	/100 pts

Please read the following carefully:

General Guidelines for Lab Assignment Submission.

- Please sign and date the statement of Academic Honesty.
- Please go to the class and lab indicated by your registered course code via the CUSIS system. If you go to a different lab than the one you are registered for, you will not receive credit for the assignment even if you completed it.
- Write your COMPLETE name and student ID number legibly on the cover sheet (otherwise we will not take any responsibility for your lab). Please write your answers using a black or blue pen, NOT any other color or a pencil.
- Write your solutions on a double-sided printout of this pdf file. Try to fit your answers inside the available space.
- The use of computers/cellular phones/graphing calculators/iPads will NOT be permitted during tests and lab assignments. Please do not use our lab computer to recharge your cellular phone battery.
- In order to make it fair for all students, during the labs and tests, if you touch/press any icons on your cellular phone, our TA will check your phone to determine whether or not you are exchanging messages with another student. If you are found cheating (in the tests or in the lab or on homework assignments), you will automatically get an F grade in this course and your act will be reported to the Department for necessary disciplinary actions.

Exercises

1. (20 marks)

Create a MATLAB function to calculate the sum of an arbitrary geometric series

$$\sum_{n=1}^N ar^{n-1},$$

when N can also be infinite (∞). There are three cases to be considered:

- (a) If $N < \infty$, then your function should calculate the series iteratively. Do not use the formula $\sum_{n=1}^N ar^{n-1} = \frac{a(1-r^N)}{1-r}$;
- (b) If $N = \infty$, and $|r| < 1$, then your function should calculate the series by the formula $\sum_{n=1}^{\infty} ar^{n-1} = \frac{a}{1-r}$;
- (c) If $N = \infty$, and $|r| \geq 1$, then your function should display “The geometric series is divergent. ” and output is empty set.

Remark: We do not consider the case where $a = 0$ (so you do not have to check whether a is 0, even for the case $N = \infty$ and $|r| \geq 1$). In addition, the output of this function is the sum of this geometric series.

Write a MATLAB function `lab8_q1(a,r,N)` to implement the above program and save it in the directory H:.

Solution for lab8_q1(a,r,N)

```
function gsum = lab8_q1(a,r,N)
% q1

if N == Inf
    if abs(r) >= 1
        disp('The geometric series is divergent.')
```

gsum = [];

```
    else
        gsum = a/(1 - r);
    end
else
    gsum = 0;
    for k = 0 : N - 1
        gsum = gsum + a*r^k;
    end
end
```

Test your function by the following choices and write down the results below.

- (a) $a = 2, r = 0.4, N = \infty$;
- (b) $a = 10, r = 1, N = \infty$;
- (c) $a = 2017, r = 1/8, N = 10$.

```
>> lab8_q1(2,0.4,inf)

ans =

    3.3333

>> lab8_q1(10,1,inf)
The geometric series is divergent.

ans =

    []

>> lab8_q1(2017,1/8,10)

ans =

    2.3051e+03
```

2. (25 marks)

Define a recursive sequence $\{x_n\}$ by

$$x_{n+1} = \frac{x_n}{25} + \frac{5}{x_n},$$

where $x_1 = 2$.

- (a) Write a function `lab8_q2a(N)` to calculate x_N by using **for-loop**. Save the function in the directory H:. Please keep using the same variable x in the iterative process. That is to say, you do not need to store the previous result ($x_n, n = 1, 2, \dots, N - 1$). The output of this function is just x_N ;

Solution for lab8_q2a.m

```
function x = lab8_q2a(n)
% q2a
x = 2;
for i = 2:n
    x = x/25 + 5/x;
end
```

- (b) Write a **recursive** function `lab8_q2b(N)` to calculate x_N by calling function itself. Save the function in the directory **H**;

Solution for lab8_q2b.m

```
function x = lab8_q2b(n)
% q3b
    if n == 1
        x = 2;
    else
        x = lab8_q2b(n-1)/25 + 5/lab8_q2b(n-1);
    end
end
```

- (c) Compare the CPU time elapsed (in seconds) for calculating the x_N in `lab8_q2a(N)` and `lab8_q2b(N)`, respectively.

Hint: You can enter the following command to record the time:

```
tic; lab8_q2a; toc
```

Write down the time record for computing x_2, x_{15} and x_{20} by using the above two functions. Try to explain why these two functions cost different time for same input x .

```
>> tic; lab8_q2a(2); toc
Elapsed time is 0.000080 seconds.
>> tic; lab8_q2b(2); toc
Elapsed time is 0.000233 seconds.
>> tic; lab8_q2a(15); toc
Elapsed time is 0.000040 seconds.
>> tic; lab8_q2b(15); toc
Elapsed time is 0.172855 seconds.
>> tic; lab8_q2a(20); toc
Elapsed time is 0.000076 seconds.
>> tic; lab8_q2b(20); toc
Elapsed time is 4.409433 seconds.
```

Because we don't need to store some previous elements in for loop program; the recursive function take more memory.

3. (25 marks)

We can represent a point either by the Cartesian coordinate or by the polar coordinate. Given a point (r, θ) in the polar coordinate, its Cartesian coordinate (x, y) is given by

$$x = r \cos \theta, \quad y = r \sin \theta,$$

where $r > 0$ and $\theta \in [0, 2\pi]$. On the other hand, when we are given a point (x, y) in the Cartesian coordinate. The formulas for expressing the same point by polar coordinate are

$$r = \sqrt{x^2 + y^2}, \quad \theta = \arctan \frac{y}{x}.$$

- (a) Write a MATLAB function `[x,y]=lab8_q3a(r,theta)` to convert a point (r, θ) represented in the polar coordinate to the Cartesian coordinate (x, y) . Your program should have 2 outputs (`x` and `y`).

Solution for lab8_q2a.m

```
function [x,y] = lab8_q3a(r,theta)
% q3a

x = r*cos(theta);
y = r*sin(theta);
```

- (b) Write a MATLAB function `[r,theta]=lab8_q3b(x,y)` to convert a point (x, y) represented in the Cartesian coordinate to the polar coordinate (r, θ) . Your program should have 2 outputs (`r` and `theta`). You should check carefully the value of θ for different cases of x and y .

Remark: This function consists of 2 inputs x and y , but not a vector with length 2.

Solution for lab8_q3b.m

```
function [r,theta] = lab8_q2b(x,y)
% q3b

r = sqrt(x^2 + y^2);
if r == 0
    theta = 0;
elseif x < 0
    theta = pi + atan(y/x);
elseif y >= 0
    theta = atan(y/x);
else
    theta = 2 * pi + atan(y/x);
end
```

Use your MATLAB function to convert the following points in Cartesian coordinate into polar coordinate.

Remark: This function has more than one output. You should run the function in the form of `[r, theta] = lab8_q3b(x,y)` instead of just `lab8_q3b(x,y)`.

- i. $(x, y) = (-1, 1)$;
- ii. $(x, y) = (2, 0)$;
- iii. $(x, y) = (9, 1)$.

```
i:    (1.4142, 2.3562)
ii:   (2, 0)
iii:  (9.0554, 0.1107)
```

4. (30 marks)

Sorting is a very important and fundamental topic in programming. Swapping is a ‘preliminary’ job in order to perform sorting.

- (a) Write a MATLAB function `swap(x,y)` to perform swapping. Let a, b be values and x, y be variables. If a is stored in x and b is stored in y originally, then after performing the function, b should be stored in x and a should be stored in y .

Solution for `swap(x,y)`

```
function [x y] = swap(x,y)
temp = x;
x = y;
y = temp;
```

- (b) Suppose a vector \mathbf{v} is given. Use the above function `swap` to write a function named as `reverse.m` for reversing \mathbf{v} . If $\mathbf{v} = (v_1, v_2, \dots, v_N)$, then after reversing, the vector becomes $\mathbf{v} = (v_N, v_{N-1}, \dots, v_1)$. Save the new function in the directory H:. Do not use the MATLAB built-in function `flip`, `fliplr`, `rot90` or `permute`.

Solution for reverse(v)

```
function [ v ] = reverse( v )
% reverse

for i = 1 : floor(length(v)/2)
[v(i) v(length(v) - i + 1)] = swap(v(i),v(length(v) - i + 1));
end
```

- (c) Suppose a vector \mathbf{v} is given. Write a function `selectionsort(v)` to perform the sorting without using the MATLAB built-in function `sort`, `find`, `min` or `max`.

Hint: You may use the selection sort algorithm. It works by first finding the minimum value in the list. Then swapping it with the value in the first position. Repeat the above steps for remainder of the list.

Solution for selectionsort(v)

```
function v = selectionsort(v)
% sorting

for i = 1 : length(v) - 1
    for j = i + 1 : length(v)
        if v(j) < v(i)
            [v(i),v(j)] = swap(v(i),v(j));
        end
    end
end
```