CSE123 - Lecture 7

Midterm Review & Functions
Midterm Review

(12 pts)
Convert the following mathematical expressions into Matlab statements:

\[ \sqrt{x^2 + y^2} \quad \text{OR} \quad (x^2+y^2)^{1/2} \]

\[ (a-b)^{1/2} \quad \text{OR} \quad \sqrt{a-b}/(c*d^3) \]

Convert the following Matlab statements into valid mathematical expressions

\[ \sqrt{\exp(2*a)}*\sin(b) \]

\[ m / (1-(v/c)^2)^{0.5} \]

Calculate the results of the following mathematical statements written in a Matlab script.

\[ \frac{1}{3} \times (144^{1/2}) \quad 24 \quad 2^3/3^2 \quad 8/9 \]

\[ 1+2^3^2 \quad 65 \quad 2e02*6 + 6^2/2 \quad 1218 \]
for i=1:3
    for j=i:3
        if(i>2)
            continue
        end
        k = i^j
    end
end

arr1(1,:) = 1:4;
arr1(2,:) = 5*ones(1,4);
[m n] = size(arr1);
for i=1:m
    for j=1:n
        arr2(i,j) = 2*arr1(i,j) - 1;
    end
end
arr2
A = eye(2,2)*3
B = (2*ones(2,2)).^2
C = [1 -1 ; 0 2]
D = C.*B
E = B*C'

A =
3 0
0 3

B =
4 4
4 4

C =
1 -1
0 2

D =
4 -4
0 8

E =
0 8
0 8
(15pts) Write an M-file that prompts the user for the type of connection (series or parallel) and the number of resistors $n$ and then computes the equivalent resistance.

```matlab
clear
clc
n = input('enter number of resistors');
ctype= input('enter type of connection parallel(p) or series(s)','s');
sum=0;
if ctype=='p';
    for i=1:n
        r=input ('enter the resistor' );
        sum=sum+1/r;
    end
    req=1/sum;
    disp('equivalent resistance is:')
    disp(req)
elseif ctype=='s';
    for i=1:n
        r=input ('enter the resistor' );
        sum=sum+r;
    end
    req=sum;
    disp('equivalent resistance is:')
    disp(req)
else
    disp('wrong type of connection ')
end
```
Midterm Review

Complete and correct the following program

a) to count and print votes of an election using a while loop and a switch block. Assume that 50 students vote for nominees A, B, C. All other blank or illegal votes will be counted as well. (15pts)

b) Then, sort the nominees in descending order with their corresponding votes. If you can not solve the part a assume you know $v_A$, $v_B$, $v_C$ and others. (15 pts)

```plaintext
n=0;
while n >50
  v=input('vote type', 's');
  switch
    case v='A'
      vA=vA+1;
    case v='B'
      vB=vB+1;
    case v='C'
      vC=vC+1;
    others
  end
  vD=vD+1
end
```

```plaintext
vA=0;vB=0;vC=0;vD=0;
n=0;
while n <50
  v=input('vote type', 's');
  switch v
    case 'A'
      vA=vA+1;
    case 'B'
      vB=vB+1;
    case 'C'
      vC=vC+1;
    otherwise
      vD=vD+1
  end
  n=n+1;
end
```

```plaintext
vA
vB
vC
vD
```
Complete and correct the following program
a) to count and print votes of an election using a while loop and a switch block. Assume that 50 students vote for nominees A, B, C. All other blank or illegal votes will be counted as well. **(15 pts)**
b) Then, sort the nominees in descending order with their corresponding votes. If you can not solve the part a assume you know \( v_A, v_B, v_C \) and others. **(15 pts)**

```matlab
A=[vA vB vC vD]
for i=1:4
    for j=1:3
        if A(j)<A(j+1)
            temp=A(j+1);
            A(j+1)=A(j);
            A(j)=temp;
        end
    end
end
A
```
1. **(25 pts)** Write a Matlab script `ans5.m` that calculates the result of the following series for a given value of $x$ with the accuracy first twenty terms.

\[
3 + x + \frac{x^3}{2!} + \frac{x^5}{3!} + \frac{x^7}{4!} \ldots
\]

```matlab
% solution for question 5
clear
clc
sum=0;x=input('enter value of x = ');for i=1:3
    f=1;
    for j=1:i
        f=f*j;
    end
    sum=sum+x^(2*i-1)/f;
end
result=sum+3;
disp(['result= ',num2str(result)])
```

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>12</td>
</tr>
<tr>
<td>Q2</td>
<td>35</td>
</tr>
<tr>
<td>Q3</td>
<td>25</td>
</tr>
<tr>
<td>Q4</td>
<td>33</td>
</tr>
<tr>
<td>Q5</td>
<td>20</td>
</tr>
<tr>
<td>Σ</td>
<td>125</td>
</tr>
</tbody>
</table>
Functions

Ex: Already defined functions: sin, cos, tan, sqrt…

Specifications

• Usually have a meaningful name (case sensitive)

  function sqrt is not the same as SQRT

• Need an argument, or parameter to work.

  sqrt(10)  sin(2*pi/180)

• Can give multiple answers or results, usually as a vector (or matrix)

  >> sqrt(10)
  ans =   3.1623

  >> SQRT(10)
  ??? Capitalized internal function SQRT;
  Caps Lock may be on.

  >> sqrt
  ??? Error using ==> sqrt
  Incorrect number of inputs.

  >> A=rand(10,10)
  >> size(A)
  ans =    10    10
User defined functions

How to create a user-defined function

- Exactly like script (.m file)
- Start with a function definition line

Typical structure of a user-defined function

a. **Function definition line** (required):
   
   ```
   function OUTPUT=NAME(INPUT)
   ```

b. **Comment Section** (information on the program)

c. **Calculation section**
   
   Unless specified, every variable is *local* and therefore will not exist after the function is applied.

d. **Output section**
   
   Needs to use the OUTPUT variable(s).
User defined functions

Same specifications

• Define the function with a **meaningful** name
• Decide how many arguments (or input)
• Decide how many results (or output)

**Ex:**

Define a function that calculates the **area of the circle** based on its **radius**.

- **Name?** Circle_Area  Area_C  Surf_C  area
- **How many input?** Only one: the radius  R
- **How many output?** Only one: the area…  A

*function OUTPUT=NAME(INPUT)*
User defined functions

function OUTPUT=NAME(INPUT)

A  area  R

function A=area(R)
A=pi*R^2;

>> area(1.2)
ans = 4.5239

>> area(1.5)
ans = 7.0686

>> area
??? Input argument 'R' is undefined.
Error in ==> area at 3
On line 3  ==> A=pi*R^2;
User defined functions

Examples of functions definitions lines:

Convert temperature from Fahrenheit to Celsius.

\[ \text{function } T\text{Cel}=\text{Far2Cel}(T\text{far}) \]

Convert angle from degrees to radians

\[ \text{function } A\text{rad}=\text{Deg2Rad}(A\text{deg}) \]

Calculate area and circumference of circle from radius

\[ \text{function } [\text{Area, Circ}]=\text{Circledata}(R) \]

Calculate the area and circumference of a rectangle

\[ \text{function } [\text{Area, Circ}]=\text{RectData}(\text{Side1, Side2}) \]

Calculate the arithmetic and the geometric mean of a series of number

\[ \text{function } [\text{Amean, Gmean}]=\text{Means}(A) \]
User defined functions

Example 1

Convert angle from degrees to radians

Define the function with a meaningful name

- Decide how many inputs: 1
- Decide how many output: 1

D2R
Adeg
Arad
User defined functions

function Arad=d2r(Adeg)
    Arad=Adeg*pi/180;

Saved as d2r.m

>> d2r(90)
ans =  1.5708
>> d2r(180)
ans =  3.1416
>> sin( d2r(90) )
ans =  1
>> cos( d2r(180) )
ans =  -1

>> disp(Adeg)
??? Undefined function or variable 'Degree'.

Local variable only exist inside the body of the function
Example 2

Function to calculate the Reynolds number of the flow in a pipe

\[ \text{Re} = \frac{(M/A)D}{\mu} \]

- \(M\): Mass flow rate.
- \(A\): Cross sectional area of the Pipe
- \(D\): Pipe diameter
- \(\mu\): Dynamic viscosity of fluid

Define the function with a meaningful name

- Decide how many inputs: 3
  - M,D,mu
- Decide how many output: 1
  - Result
function Result=Re1(M,D,mu)
A=pi*(D/2)^2;
Result=(M/A)*D/mu;

>> Re1(1,2, 9.7720E-4 )
ans =651.4734

>> disp(A)
??? Undefined function or variable 'A'.

>> disp(Result)
??? Undefined function or variable 'Result'.

Local variables only exist inside the body of the function
function Result=Re2(M,D,q)
% q values: (1) for water and (2) for air
A=pi*(D/2)^2;
if q==1
    mu= 9.7720E-4 ;
else
    mu= 1.8205E-5;
end
Result=(M/A)*D/mu;

All variables inside the body of the function are local variables

Saved as Re2.m

>> Re2(1,2,1)
ans =  651.4734

>> Re2(1,2,2)
ans =  3.4970e+004
function Result=Re3(M,D)

global mu

A=pi*(D/2)^2;

Result=(M/A)*D/mu;

mu needs to be defined inside and outside as a global variable.

>> global mu
>> mu= 9.7720E-4

>>Re3(1,2)
ans = 651.4734
Example for Functions

Distance between two points

```matlab
function distance=dist2(x1,y1,x2,y2)
  %DIST2 calculate the distance between two points
  distance= sqrt((x2-x1).^2+(y2-y1).^2);
```

A simple script calling this function test_dist2.m

```matlab
disp('calculate the distance between two points')
ax=input('enter x value of first point');
ay=input('enter y value of second point');
bx=input('enter x value of first point');
by=input('enter y value of second point');
result=dist2(ax,ay,bx,by);
disp(['the distance between two points is :', num2str(result)]);
```