

შესავალი

MATLAB მეხსიერება:

Variables
Functions

MATLAB ცვლადები:

Scalar
Vector
Matrix (vector)

```
>> A = 10;
>> B = zeros(1,A);
>> C = zeros(A,1);
>> D = zeros(A,A);
>> E = zeros(A,A,A);
>>
>> whos
Name          Size          Bytes  Class
A             1x1           8     double array
B             1x10          80    double array
C            10x1           80    double array
D            10x10         800   double array
E            10x10x10     8000  double array

Grand total is 1121 elements using 8968 bytes

>>
```

MATLAB Output:

```
>> A = zeros(1,10);  
>> B = zeros(1,10)  
  
B =  
  
    0    0    0    0    0    0    0    0    0    0  
  
>>
```

MATLAB Memory:

შეაფასეთ მაქსიმალური ზომის ცვლადი თქვენი კომპიუტერისათვის (Ram 1 Gb):

```
>> M=1e+9;    % Memory = 1Gb, M=8*N^3  
>>  
>> N = (M/8)^(1/3)  
  
N =  
  
    500.0000  
  
>>
```

3 განზომილებიანი მატრიცის მაქსიმალური სიდიდე: 500x500x500

```
>> whos
```

დავიწყოთ სუფთა ფურცლიდან:

```
>> clear    % CLEAR VARIABLES  
>> clear all % CLEAR VARIABLES AND FUNCTIONS  
>> clc     % CLEAR COMMAND WINDOW
```

Using Functions

Matlab Workspace / main file

```
A1 = 0;  
A2 = 1;  
  
[B1, B2] = tempf(A1, A2);
```

Current_Folder\tempf.m

```
function [out1, out2] = tempf(in1, in2);  
  
local_variable = 5;  
  
out1 = 0;  
out2 = 0;
```

მიმდინარე ფოლდერის გაგება:

```
pwd
```

print working directory (Unix)

ფოლდერის დამატება:

```
addpath('c:/matlab/mymatlabfiles/lecture1');
```

Arithmetic operations

```
>> A = 10;
>> B = zeros(1,A);
>> C = zeros(A,1);
>> D = zeros(A,A);
>> E = zeros(A,A,A);
>>
>>
>>
>> A1 = A + 5 + 5*A + A/5 + A^5 + sin(A);
>> % SCALAR OPERATIONS
>>
>> B1 = B + 5*B + B.*B + B./B + B.^5 + sin(B);
>> % VECTOR OPERATIONS
>>
>> C1 = C + B';
>> % TRANSPOSE
>>
```

ოპერაციები ორ ვექტორზე: `.*` `./` `.^`

ვექტორების გამრავლებისას მათ უნდა გააჩნდეთ ერთიდაიგივე განზომილება;

```
Error using *
Inner matrix dimensions must agree.
```

`whos`

Logics

sample.m

```
for ind = 1:10
    T(ind)=ind;
    Blabla=1;
    Blabla=2;
end
```

Graphics

plotting.m

```
for ind=1:1:10
    T(ind)=ind;
end

figure(1);
plot(T,sin(T));
```



ფაილის სახელი არ უნდა ემთხვეოდეს მატლაბის სტანდარტული ფუნქციის სახელს.
მაგ. `plotting.m` და არა `plot.m`

Using MATLAB Help



როგორ მუშაობონ გრაფიკული ფუნქციები:

plot
subplot

xlabel
ylabel

hold on
hold off

comet

დავალეზა 1.1.

```
x=(0.1:0.1:10); % vector of 0.1->10 with step 0.1
```

დახაზეთ ერთ გრაფიკზე:

$100 \sin(x)/x$	(წითელი)
x^3	(ლურჯი)
x^2	(მწვანე)

დავალბა 1.2.

მიმდინარე დირექტორიაში შექმენით ფუნქცია phases:

```
phases.m
```

```
function [a1, a2] = phases(x);  
  
a1 = sin(0.9*x);  
a2 = cos(1.1*x);
```

დაწერეთ სკრიპტი `pirveli.m` რომელიც:

- განსაზღვრავს ვექტორს: `x=(0:0.1:25)` ;
- `a1` და `a2` ვექტორების გამოსათვლელად გამოიყენებს ფუნქციას `phases` ;
- პირველ ნახაზზე `figure(1)` დახაზავს `a1(x)`, `a2(x)` ერთ გრაფიკზე ;
- მეორე ნახაზზე დახაზავს `a1(a2)` `subplot(1,2,1)`-ზე და `a2(a1)` `subplot(2,1,2)`-ზე ;
- მესამე ნახაზზე მოახდენს `a1(a2)` გრაფიკის ანიმაციას ფუნქციით `comet` ;

Vector Patch

```
>>
>> t = (0:1:100);
>>
>> F = sin(t).^2;
>>
>> figure(1);
>> plot(t(1:50),F(1:50));
>>
>> whos
  Name      Size      Bytes  Class
  ----      -
  F         1x101     808    double array
  t         1x101     808    double array

Grand total is 202 elements using 1616 bytes

>>
```

plot ფუნქციაში x/y მიმართულების ვექტორებს უნდა გააჩნდეს ერთიდაიგივე ელემენტების რაოდენობა.

```
Error using plot
Vectors must be the same lengths.
```


Tensor Patches

```
>>
>> A = ones(100,100);
>> B = A(3,:);
>> C = A(:,21);
>> whos
  Name      Size      Bytes  Class
  A         100x100    80000  double array
  B           1x100     800    double array
  C         100x1     800    double array
  F           1x101     808    double array
  t           1x101     808    double array

Grand total is 10402 elements using 83216 bytes

>>
```

Tensor Length

```
>>
>> A = ones(10,20);
>> N1 =length(A(1,:));
>> N2 = length(A(:,1));
>>
>> N1

N1 =

    20

>> N2

N2 =

    10

>> whos
  Name      Size      Bytes  Class
-----
  A         10x20     1600   double array
  N1         1x1         8     double array
  N2         1x1         8     double array

Grand total is 202 elements using 1616 bytes

>>
```



როგორ მუშაობენ ფუნქციები:

sum
diff
rand

ინტეგრალის შეფასება დისკრეტული ჯამით

$$Int = \sum_i^N (f_i \Delta t_i) = \Delta t \sum_i^N f_i$$

```
clear all           % Start

N = 25;            % Sampling rate
dt = pi/(N-1);    % Time step
t = (0:dt:pi);    % Time vector with N elements

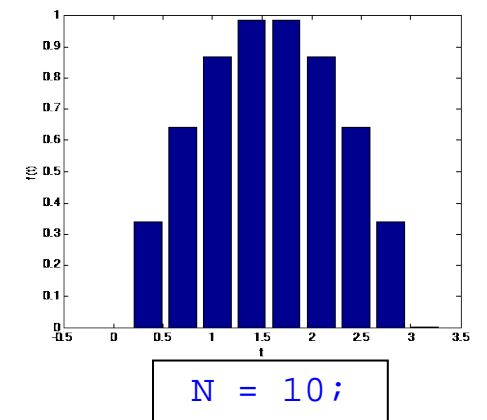
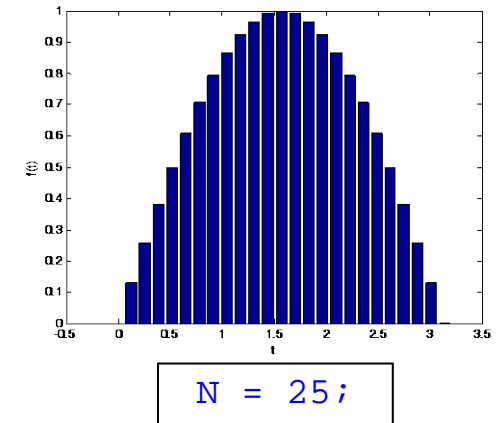
f = sin(t);       % Function to integrate

figure(1);
bar(t,f);
xlabel('t');
ylabel('f(t)');

Int = sum(f)*dt    % Integration through summation
```

```
Int =
    1.9971

>>
```



წარმოებულის შეფასება დისკრეტული სხვაობებით

$$\left(\frac{df}{dt}\right)_i = \frac{f_{i+1} - f_i}{t_{i+1} - t_i} = \frac{\Delta f_i}{\Delta t}$$

```
clear all %START

N = 25;           % Sampling rate
dt = pi/(N-1);   % Time step
t = (0:dt:pi);   % Time vector with N elements

f = sin(t);      % Function to differentiate
dfdt = diff(f)/dt; % Primitive derivative through difference

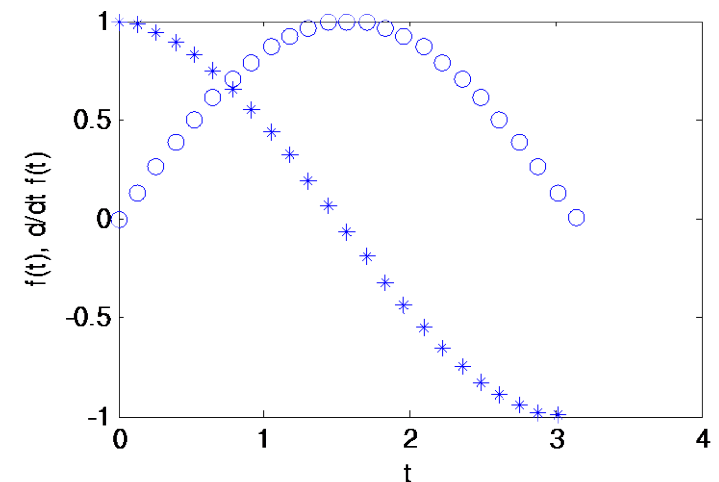
figure(1);
plot(t,f,'o');
hold on
plot(t(1:N-1),dfdt,'*');
hold off

xlabel('t');
ylabel('f(t), d/dt f(t)');
```

```
>> whos
Name      Size      Bytes  Class

N         1x1        8      double
dfdt      1x24      192     double
dt        1x1        8      double
f         1x25      200     double
t         1x25      200     double

>>
```



დავალება 1.3.

- განსაზღვრეთ ვექტორი: $t = (0:0.1:10)$;
- გამოითვალეთ ფუნქცია: $F(t) = \exp(\sin(t^2))$;
- გამოითვალეთ ფუნქცია: $G(t) = \tan(\sin(t^2))$;
- პირველი ნახაზის ქვეგრაფიკზის პოზიციაზე (2,2,1) დახაზეთ $F(t)$ ფუნქციის დამოკიდებულება t -ზე ინტერვალში 0-დან 5-მდე.
- პირველი ნახაზის ქვეგრაფიკზის პოზიციაზე (2,2,2) დახაზეთ $G(t)$ ფუნქციის დამოკიდებულება t -ზე ინტერვალში 0-დან 8-მდე.
- გამოითვალეთ ფუნქცია $H(t) = \int_0^t F(t) dt$ მარტივი აჯამვის გამოყენებით (`sum`).
- მეორე გრაფიკზე დახაზეთ ფუნქცია $H(t)$ -ს დამოკიდებულება t -ზე ინტერვალში 0-დან 5-მდე.
- გამოითვალეთ ფუნქცია $H_1(t) = H(t)\cos(t) + 0.01*(\text{random number } [0-1])$;
- გამოითვალეთ ფუნქცია $H_2(t) = d/dt H_1(t)$ მარტივი სხვაობების გამოყენებით (`diff`) ;
- დახატეთ ერთ გრაფიკზე ინტერვალში 0-დან 8-მდე:

$F(t)$	(შავი)
$H(t)$	(წითელი)
$H_1(t)$	(მწვანე)
$H_2(t)$	(ლურჯი)

Gradient

$$\nabla F = \frac{\partial F}{\partial x} \hat{i} + \frac{\partial F}{\partial y} \hat{j}$$

$$\nabla F = \frac{\partial F}{\partial x} \hat{i} + \frac{\partial F}{\partial y} \hat{j} + \frac{\partial F}{\partial z} \hat{k} + \dots$$



gradient
contour
quiver

linspace

დავალეზა 1.4.

- განსაზღვრეთ ვექტორები: `x = linspace(0,2,100); y = linspace(0,2,100);`
- გამოთვალეთ ფუნქციის გრადიენტი ორ განზომილებაში: $F(x,y) = x \cdot \exp(-x^2 + y^2)$;
- ააგეთ ველის გრადიენტის $grad F(x,y)$ კონტურული ნახაზი (`contour`);
- ააგეთ ველის გრადიენტის $grad F(x,y)$ ვექტორული ნახაზი (`quiver`);

