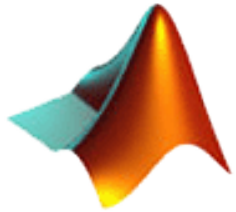
MATLAB



جلسه سوم

Hossein Rezaei (hrezaei@ut.ac.ir)

جلسه سوم :

فهرست مطالب :

۱ - یادآوری جلسه دوم

۲- ایجاد توابع در متلب

۳- بررسی رسم توابع دو بعدی در متلب

۱- یادآوری جلسه دوم

❖ ساختارهای کنترلی if و for

❖**مثال** : برنامه‌ای که با استفاده از ساختارهای کنترلی تودرتو، ماتریسی با ابعاد دلخواه را دریافت نماید و حاصل جمع روی قطر اصلی آن را محاسبه کند.

۱- یادآوری جلسه دوم

```
clc ;
clear ;
close all ;

A = input ('Matrix A=') ;
sum = 0 ;
[r c] = size(A) ;
for i = 1 : r
        for j = 1 : c
                if  i == j
                        sum = sum + A(i,j);
                end
        end
end
```

۱- یادآوری جلسه دوم

❖ بررسی همزمان چند شرط در ساختار کنترلی Switch

❖مثال : برنامه‌ای که روزهای زوج و فرد هفته را مشخص می‌کند.

۱- یادآوری جلسه دوم

```
num = input('Enter a number = ');

switch num
    case {1,3,5}
        disp ('This day is an Even weekday.');

    case {2,4,6}
        disp ('This day is an Odd weekday.');

    case 7
        disp ('This day is not Even or Odd.');

end
```

۱- یادآوری جلسه دوم

❖ساختار کنترلی Switch

❖مثال: استفاده از رشته‌ها در دستورات شرط Switch و ایجاد یک رابط گرافیکی برای نمایش پیغام (questdlg)

۱- یادآوری جلسه دوم

```
ANSWER= questdlg('Which programming language do you use?','Question', ...
    'MATLAB','C/C++','VB','MATLAB');

switch ANSWER
    case 'MATLAB'
        disp(' MMMMM');

    case 'C/C++'
        disp('CCCCC');

    case 'VB'
        disp('VVVVV');
end
```

۲- ایجاد توابع در متلب

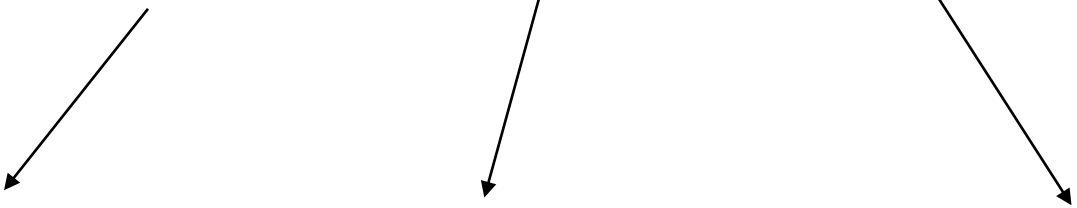
با استفاده از توابع در متلب، می‌توانیم یک بار یک کد را بنویسیم و چندین بار از آن استفاده کنیم.

مزایای استفاده از توابع به جای فایل‌های اسکریپت :

- سرعت بالاتر
- صرفه‌جویی در حافظه کامپیوتر
- توسعه توانایی‌های متلب

۲- ایجاد توابع در متلب

function [*outputs*] = function_name (*inputs*)



Output variables

Must match the file name

input variables

۲- ایجاد توابع در متلب

سوال : تابعی بنویسید که ۳ عدد را از ورودی گرفته و جمع و ضرب آنها را در خروجی نمایش دهد.

۲- ایجاد توابع در متلب

```
clc ;
clear ;
close all ;

A = input ('Enter First Number : ') ;
B = input ('Enter Second Number : ') ;
C = input ('Enter Third Number : ') ;

numadd (A,B,C) ;
nummul (A,B,C) ;

disp([' A + B + C = ', numadd]);
disp([' A * B * C = ', nummul]);
```

۲- ایجاد توابع در متلب

```
function s = numadd (A,B,C) ;
      s = A + B + C;
end
```

```
function m = nummul (A,B,C) ;
      m = A * B * C;
end
```

۲- ایجاد توابع در متلب

```
function [s m] = numResult (A,B,C) ;
      s = A + B + C;
      m = A * B * C;
end
```

۲- ایجاد توابع در متلب

سوال : تابعی بنویسید که ماتریسی با ابعاد دلخواه کاربر را دریافت نماید و حاصل جمع روی قطر اصلی آن و میانگین ماتریس را محاسبه کند و نمایش دهد.

۲- ایجاد توابع در متلب

```
clc ;
clear ;
close all ;

A = input (‘Enter a Matrix : ') ;
Disp(['Result 1 = ', mprol (A)]);
Disp(['Result 2 = ', avrage (A)]);
```

۲- ایجاد توابع در متلب

```
function [sum, avrage] = mprol(A)
        sum = 0;
        [r c] = size(A);
        for i = 1:r
                for j = 1:c
                        if i==j
                                sum = sum + A(i,j);
                        end
                end
        end
         avrage = mean(mean(A));
end
```

۲- ایجاد توابع در متلب

سوال : تابعی بنویسید که ۲ عدد از ورودی گرفته شود و مقادیر آنها با همدیگر عوض شود.

۲- ایجاد توابع در متلب

```
clc ;
clear ;
close all ;

A = input (‘Enter First Number : ') ;
B = input (‘Enter Second Number : ') ;

Disp([‘num1 = ', num2str(A)]);
Disp([‘num2 = ', num2str(B)]);

swap(A,B);

Disp([‘num1 = ', num2str(A)]);
Disp([‘num2 = ', num2str(B)]);
```

۲- ایجاد توابع در متلب

```
function [ a , b ] = swap ( A , B )
    temp = A;
    A = B;
    B = temp;
end
```

نکات

نکته ۱: بهتر است در هنگام نوشتن یک برنامه آنرا بصورت اسکریپت بنویسیم تا اشکالزدایی آن آسانتر باشد اما پس از کامل شدن برنامه آنرا به فانکشن تبدیل کنیم تا سرعت و کیفیت آن افزایش یابد.

نکته ۲: محیط کاری فایل‌های متنی همان محیط کاری متلب است اما محیط کاری هر تابعی مختص خود اوست یعنی اگر متغیری در یک تابع تعریف شود، تنها در آن تابع قابل دسترسی است و برعکس متغیرهای تعریف شده در محیط کاری متلب در داخل توابع تعریف شده نیستند. (مگر اینکه بصورت عمومی تعریف شده باشند)

نکات

نکته ۳: تابع ممکن است هیچ آرگومان ورودی یا خروجی نداشته باشد.

۳- بررسی رسم توابع دو بعدی در متلب

- چرا ترسیم داده‌ها برای ما مهم هستند؟

  – در هر زمینه علمی، ما غالباً مسأله‌ای را حل می‌کنیم و شبیه‌سازی‌هایی را انجام می‌دهیم و با توجه به اینکه غالباً تمام داده‌هایمان را نمی‌توانیم در قالب جدول بیاوریم و ترسیم داده‌ها به صورت نمودار راه‌حل مناسبی برای نمایش نتایج است.

۳- بررسی رسم توابع دو بعدی در متلب

- دستورات رسم توابع در متلب
  1. دستور Plot
     - دستور linspace
     - دستور grid
     - دستور xlim
     - دستور ylim

مثالی ساده از دستور plot

```
x = 0:0.1:10;
y = x.^2;
plot(x,y);
```

۳- بررسی رسم توابع دو بعدی در متلب

مثال ۱:

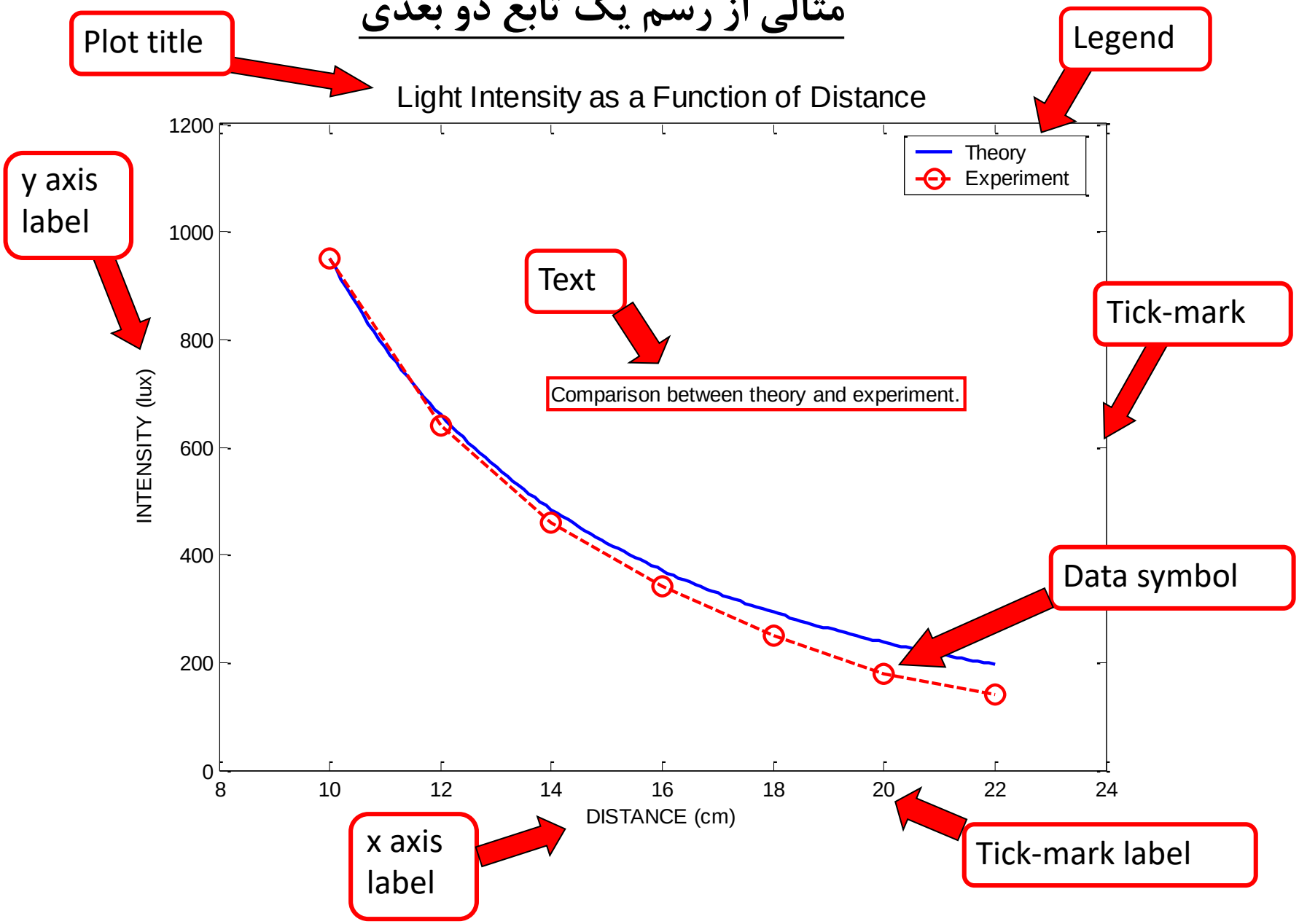
الف) برنامه‌ای بنویسید که ۵۰۰۰ عدد را در بازه (4π , 4π -) گرفته و سینوس آنها را به صورت نمودار نمایش دهد.

۳- بررسی رسم توابع دو بعدی در متلب

حل:

```
xmin=-4*pi;
xmax= 4*pi;

x=linspace(xmin,xmax,5000);
y=sin(x);


plot(x,y);
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
```

مثالی از رسم یک تابع دو بعدی



By : Hossein Rezaei (hrezaei@ut.ac.ir)

۳- بررسی رسم توابع دو بعدی در متلب

مثال ۲:

برنامه‌ای بنویسید که سینوس و کسینوس داده‌های مثال قبل را، به صورت نمودار نمایش دهد و عملیات زیر را بر روی آن انجام دهید.

- **عملیات روی نمودار :**
  1. **ترسیم دو تابع به صورت جداگانه (figure)**
  2. **ترسیم دو تابع در کنار هم (hold)**
  3. **تغییر رنگ نمودارها**
  4. **تغییر شکل خطوط نمودار**
  5. **نوشتن توضیح برای نمودار (legend)**
  6. **نوشتن عنوان برای نمودار، محور افقی و محور عمودی**

۳- بررسی رسم توابع دو بعدی در متلب

حل:

- قسمت مشترک تمامی آیتم‌های اسلاید قبل:

```
xmin=-4*pi;
xmax= 4*pi;
x=linspace(xmin,xmax,2000);

y1=sin(x);
y2=cos(x);
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۱)

```
figure;
plot(x,y1);
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);

figure;
plot(x,y2);
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۲)

```
figure;
plot(x,y1);

hold on;
plot(x,y2);
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('sin(x)','cos(x)');
hold off;
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۳)

```
figure;
plot(x,y1,'b');

hold on;
plot(x,y2,'r');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('sin(x)','cos(x)');
hold off;
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۴)

```
figure;
plot(x,y1,'-ko','LineWidth',5);
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);

hold on;
plot(x,y2,'-b*');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۵)

```
figure;
plot(x,y1,'b');

hold on;
plot(x,y2,'r');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('sin(x)','cos(x)');
hold off;
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: گزینه (۶)

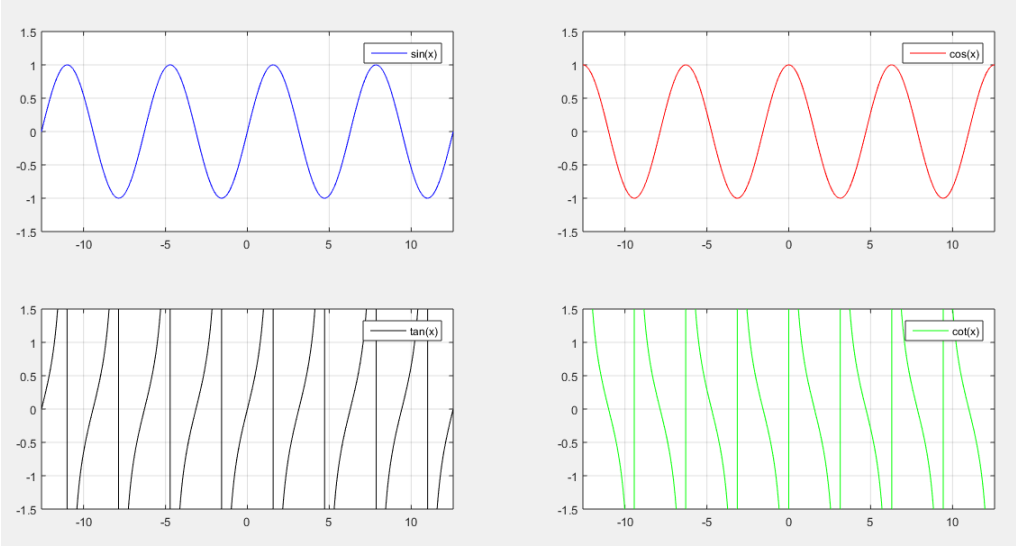
```
figure;
plot(x,y1,'b');

hold on;
plot(x,y2,'r');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('sin(x)','cos(x)');
title('x-y plot');
xlabel('x');
ylabel('y');
```

۳- بررسی رسم توابع دو بعدی در متلب

مثال ۳:

الف) برنامه‌ای بنویسید که sin و cos و tan و cot داده‌های مثال قبل را، در یک شکل و در کنار هم نمایش دهد.

ب) سپس نمودارها را با فرمت مطلوبی ذخیره کنید.



۳- بررسی رسم توابع دو بعدی در متلب

حل: (الف)

- بخش اول تولید داده های اولیه است.

```
xmin=-4*pi;
xmax= 4*pi;

x=linspace(xmin,xmax,2000);

y1=sin(x);
y2=cos(x);
y3=tan(x);
y4=cot(x);
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: (ادامه ۱)

```
figure;
subplot(2,2,1);
plot(x,y1,'b');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('sin(x)');

subplot(2,2,2);
plot(x,y2,'r');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('cos(x)');
```

۳- بررسی رسم توابع دو بعدی در متلب

حل: (ادامه ۲)

```
subplot(2,2,3);
plot(x,y3,‘k');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend(‘tan(x)');

subplot(2,2,4);
plot(x,y4,‘g');
grid on;
xlim([xmin xmax]);
ylim([-1.5 1.5]);
legend('cot(x)');
```

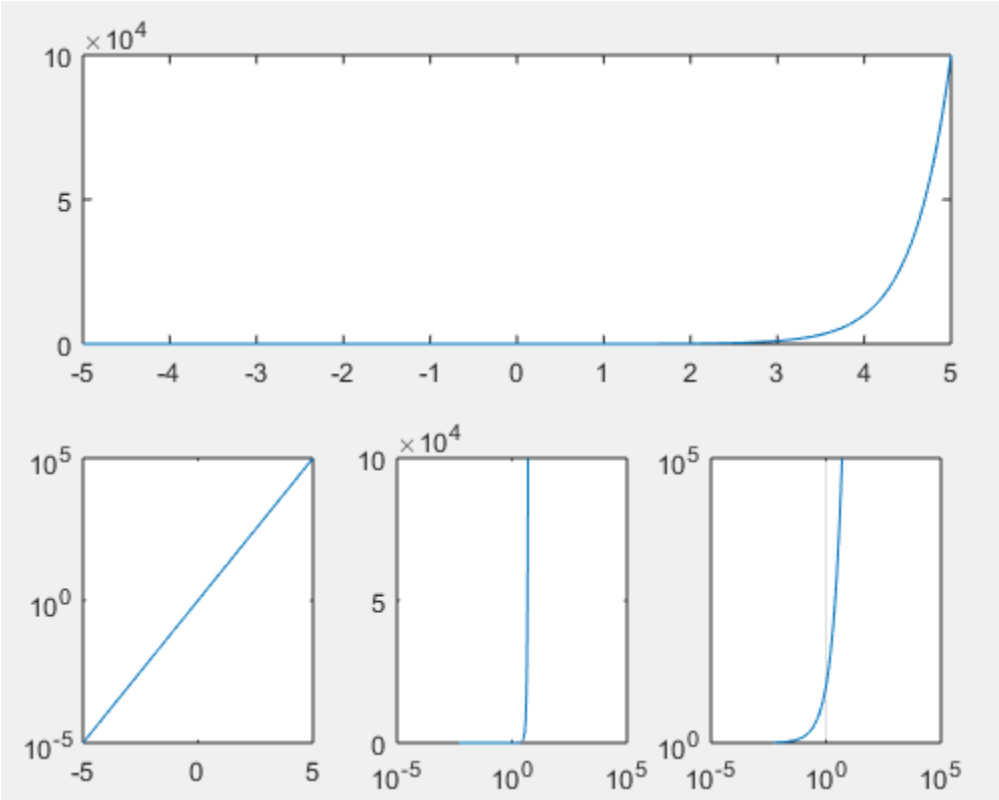
۳- بررسی رسم توابع دو بعدی در متلب

ب)

- فرمت مناسب برای ذخیره کردن نمودارها، فرمت **emf** است.
- فرمت **fig** فقط در محیط متلب باز می شود.

۳- بررسی رسم توابع دو بعدی در متلب

مثال ۴:



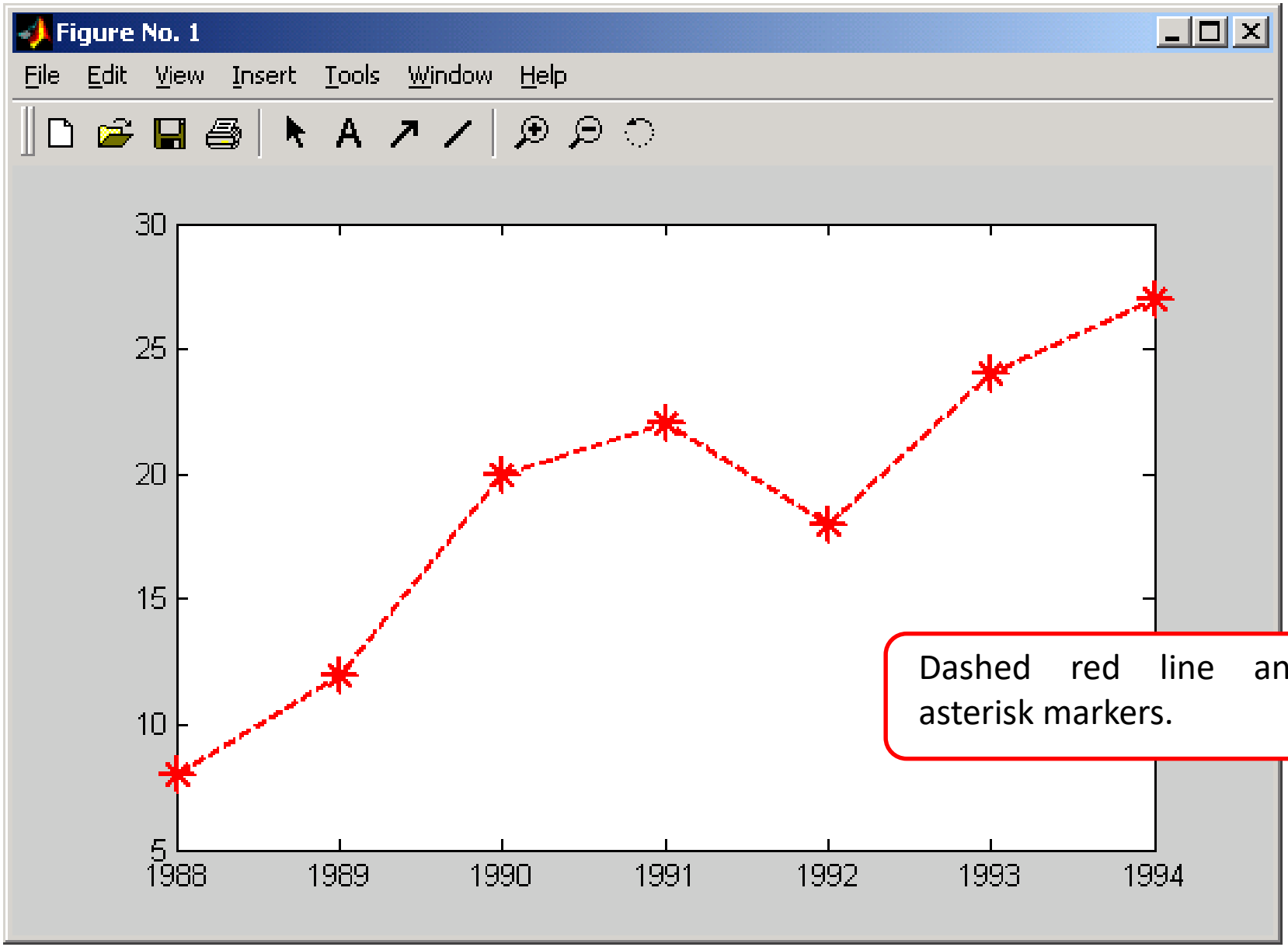
مثال ۵:

رسم نمودار داده‌های زیر:

Year	1988	1989	1990	1991	1992	1993	1994
Sales (M)	127	130	136	145	158	178	211

```
year = [1988:1:1994];
sales = [127, 130, 136, 145, 158, 178, 211];
plot(year,sales,'--r*')
```

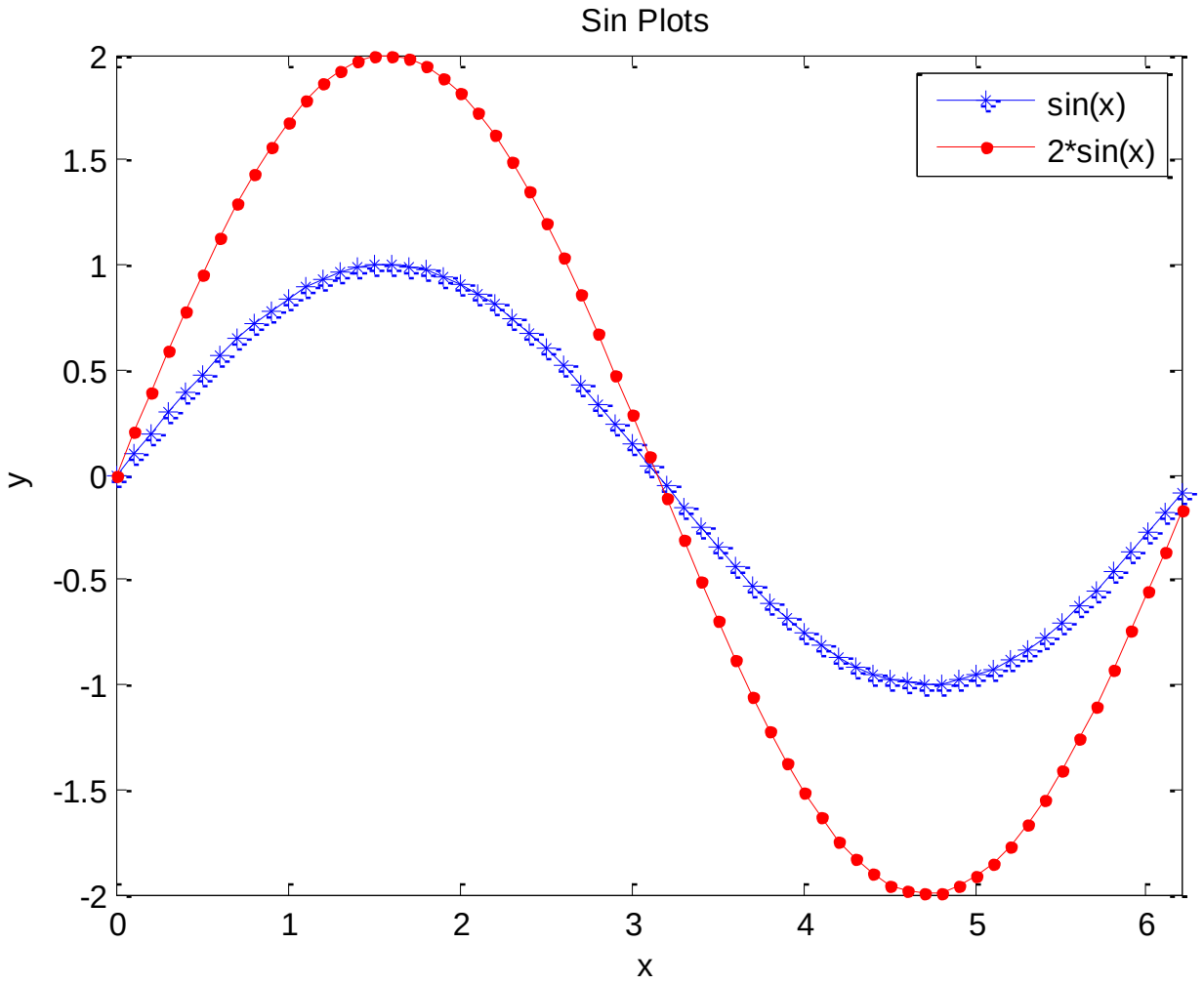
Line Specifies:
dashed red line and
asterisk markers.



By : Hossein Rezaei (hrezaei@ut.ac.ir)

مثال ۶:

رسم نمودار داده‌های زیر:



مثال ۶:

رسم نمودار داده‌های زیر:

```
x = [0:0.1:2*pi];
y = sin(x);

plot(x, y, 'b*-')

hold on
plot(x, y*2, ‘r.-')
title('Sin Plots');

legend('sin(x)', '2*sin(x)');
xlabel(‘x’);
ylabel(‘y’);
hold off
```

۳- بررسی رسم توابع دو بعدی در متلب

سایر دستورات رسم توابع دو بُعدی:

1. دستور stairs
2. دستور area
3. دستور stem
4. دستور bar
5. دستور ezplot
6. دستور comet
7. و ....

۴- بررسی رسم توابع سه بعدی در متلب

- برخی توابع رسم توابع سه بعدی در محیط متلب
  1. تابع plot3
  2. تابع mesh
  3. تابع surf
  4. تابع surfc
  5. تابع surfl
  6. تابع meshz
  7. تابع waterfall
  8. و...

۴- بررسی رسم توابع سه بعدی در متلب

مثال :

- برنامه‌ای بنویسید که با استفاده از دستور **plot3** نمونه‌ای از رسم تابع سه بعدی در متلب را نشان دهد.

۴- بررسی رسم توابع سه بعدی در متلب

حل:

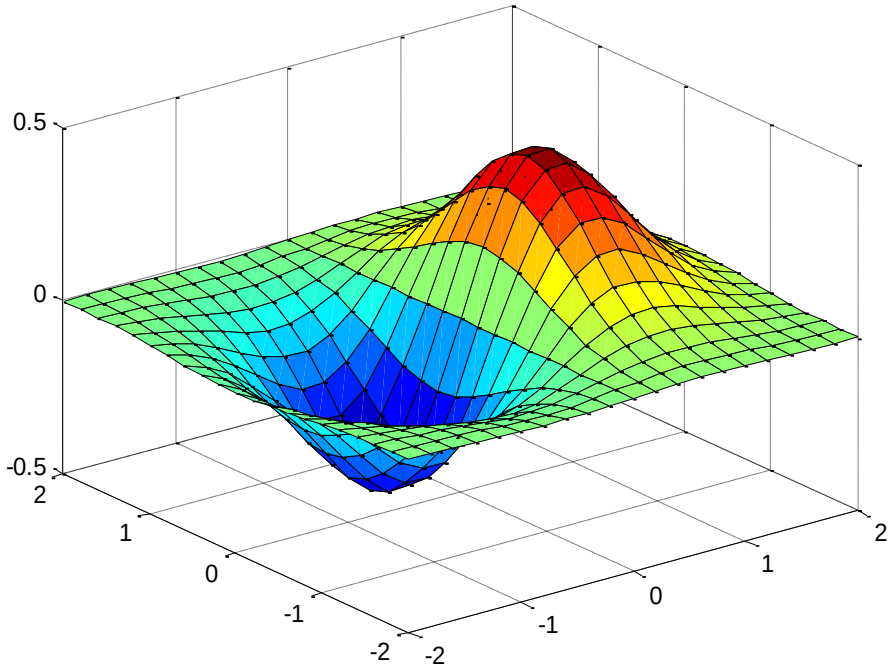
```
clc;
clear;
close all;

t = 0 : 0.01 : 10;
X = 3 * t;
Y = 4 * t;
Z = 25 - (t-5) .^ 2;

figure;
plot3(x,y,z);
grid on;
```

۴- بررسی رسم توابع سه بعدی در متلب

مثال :

- برنامه‌ای بنویسید که در نمونه‌ای از رسم تابع سه بعدی در متلب را نشان دهد.
  - دستور **meshgrid** برای ساخت ماتریس‌های دو بعدی برای رسم شکل‌های سه بعدی

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);
Z = X .* exp(-X.^2 - Y.^2);
surf(X,Y,Z)
```



۵- کار با فایل‌ها در متلب

در متلب با فرمت‌های مختلفی از فایل‌ها می‌تواند کار کرد.

- نرم‌افزار متلب یکی از بهترین نرم‌افزارها در مورد کار با فایل‌ها است که امکانات بی‌نظیری را در اختیار کاربران قرار می‌دهد.

۵- کار با فایل‌ها در متلب

- نحوه کار بر روی فایل‌های اکسل :
  - تابع xlsread
  - تابع xlswrite

۵- کار با فایل‌ها در متلب

- تابع xlswrite :

```
values = {1, 2, 3 ; 4, 5, 'x' ; 7, 8, 9};
headers = {'First','Second','Third'};
xlswrite('myExample.xlsx',[headers; values]);
```

سوال ؟