



دانشگاه صنعت آب و برق

جزوه آموزشی Matlab

- محاسبات عددی -

محمد سرفراز

فهرست

۸.....	مقدمه
۹.....	دستورات پاک کردن
۹.....	توابع مثلثاتی
۱۰.....	توابع نمایی
۱۰.....	توابع گرد کردن
۱۰.....	توابع ریاضیات گسسته
۱۱.....	توابع اعداد مختلط
۱۳.....	فرمت‌ها
۱۴.....	معرفی تابع یک متغیره
۱۵.....	رسم در مختصات قطبی
۱۵.....	رسم توابع یک متغیره با ezplot
۱۸.....	معرفی تابع دو متغیره
۱۹.....	عملیات بر روی چند جمله‌ای‌ها
۲۱.....	نمایش با دستور pretty
۲۱.....	محاسبه حدود توابع
۲۲.....	محاسبه مشتق توابع
۲۳.....	محاسبه انتگرال معین
۲۴.....	محاسبه انتگرال نامعین
۲۵.....	محاسبه انتگرال دو گانه
۲۶.....	محاسبه سری

محاسبه سری تیلور توابع یک متغیره	۲۷
پیدا کردن ریشه‌های معادله $f(x) = 0$	۲۷
حل دستگاه معادلات	۳۲
رسم بردار در فضای دو بعدی	۳۳
رسم بردار در فضای سه بعدی	۳۴
رسم صفحه	۳۴
مشخص کردن بردارهای کنج فرنه (T&N&B)	۳۵
ترسیم توابع عددی دو متغیره	۳۷
ترسیم توابع دو متغیره با مختصات قطبی	۳۹
منحنیهای تراز	۴۱
تکنیک رسم سریع توابع دو متغیره	۴۱
رسم بردار گرادیان و منحنی تراز	۴۲
رسم سطوح تراز توابع سه متغیره	۴۳
ترسیم میدان برداری گرادیان	۴۵
ترسیم پارامتریک سطوح	۴۷
استفاده از فرمان ezsurf برای ترسیم سطوح پارامتری	۴۹
معادلات دیفرانسیل مرتبه اول	۵۰
حل معادلات دیفرانسیل با شرایط مرزی	۵۰
معادلات دیفرانسیل مرتبه دوم	۵۳
معادلات مرتبه n -ام	۵۴
معادله دیفرانسیل لژاندر	۵۵
معادله دیفرانسیل بسل	۵۵

- ۵۶..... تابع بسل نوع اول
- ۵۶..... تابع بسل نوع دوم
- ۵۶..... تابع بسل نوع سوم
- ۵۷..... دستگاه معادلات دیفرانسیل خطی
- ۵۸..... تبدیل لاپلاس
- ۵۹..... تبدیل لاپلاس معکوس
- ۶۰..... تابع گاما
- ۶۰..... تبدیل فوریه
- ۶۱..... تبدیل فوریه معکوس
- ۶۲..... وارد کردن ماتریس
- ۶۲..... ترانزاده ماتریس
- ۶۲..... دترمینان ماتریس
- ۶۳..... اثر (trace) ماتریس
- ۶۳..... معکوس ماتریس
- ۶۳..... عملیات روی ماتریس ها
- ۶۴..... تولید ماتریس واحد، قطری، یک و صفر
- ۶۶..... تولید ماتریس با درایه های تصادفی توزیع یکنواخت
- ۶۶..... تولید ماتریس با درایه های تصادفی توزیع نرمال
- ۶۷..... اندیس ماتریس
- ۶۷..... تولید ماتریس با علامت : (colon)
- ۶۹..... عملیات روی سطر و ستون ماتریس
- ۷۰..... مشاهده سائز ماتریس

۷۰	تابع sum
۷۱	تابع repmat
۷۱	تولید بردار با رابطه خطی
۷۲	تولید بردار با رابطه لگاریتمی
۷۲	حذف درایه‌های ماتریس
۷۳	عملیات بر روی آرایه‌ها
۷۴	رسم نمودار توابع یک متغیره با دستور plot
۷۶	رسم پارامتریک منحنی در صفحه با plot
۷۷	رسم پارامتریک منحنی در فضا با plot3
۷۷	دستور reshape
۷۸	M-File ها
۷۹	function M-File
۸۰	دستور feval
۸۰	استفاده از عبارات منطقی برای محاسبه توابع چند ضابطه‌ای
۸۱	script M-File
۸۳	حل معادلات غیر خطی به روش نصف کردن فاصله‌ها
۸۵	حل معادلات غیر خطی به روش درونیابی خطی
۸۶	حل معادلات غیر خطی به روش درونیابی خطی اصلاح شده
۸۶	حل معادلات غیر خطی به روش تکرار تابعی
۸۷	حل معادلات غیر خطی به روش نیوتن
۸۹	حل معادلات غیر خطی به روش مولر
۹۰	حل دستگاه معادلات خطی به روش حذفی گاوس - جردن

- حل دستگاه معادلات خطی به روش تجزیه ماتریس به L و U ۹۲
- حل دستگاه معادلات خطی به روش تجزیه چولسکی ۹۴
- حل دستگاه معادلات خطی به روش ژاکوبی ۹۵
- حل دستگاه معادلات خطی به روش گاوس - سیدل ۹۷
- حل دستگاه معادلات غیر خطی به روش نیوتن ۹۹
- محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با دستور eig ۱۰۱
- محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش لوریه - فادیو ۱۰۲
- محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش توانی ۱۰۳
- محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش تکراری معکوس ۱۰۴
- درون یابی به وسیله چند جمله ای های لاگرانژ ۱۰۶
- درون یابی به روش نیویل ۱۰۷
- درون یابی به روش تفاضل محدود ۱۰۸
- درون یابی به روش حداقل مربعات ۱۰۹
- انتگرال گیری عددی به روش ذوزنقه ای ۱۰۹
- انتگرال گیری عددی به روش 13 سیمپسون (سیمپسون مرکب) ۱۱۰
- انتگرال گیری عددی به روش رامبرگ ۱۱۰
- حل انتگرال دو گانه بر روی ناحیه مستطیلی با روش سیمپسون ۱۱۱
- حل انتگرال سه گانه بر روی ناحیه مکعب مستطیلی با روش سیمپسون ۱۱۱
- حل معادلات دیفرانسیل معمولی به روش اوایلر ۱۱۲
- حل معادلات دیفرانسیل معمولی به روش هیون ۱۱۲
- حل معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه دوم ۱۱۳
- حل معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه چهارم ۱۱۳

حل معادلات دیفرانسیل معمولی به روش میلن سیمپسون ۱۱۴

حل دستگاه معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه چهارم ۱۱۵

مقدمه

Matlab یک زبان با کاربری فوق العاده برای عملیات محاسباتی است، که امکانات متعدد محاسباتی، نمایشی و برنامه‌نویسی را در محیطی که استفاده از آن برای کاربر آسان است، فراهم می‌کند. در این محیط برای مسائل و پاسخها از علائم و سمبلهای آشنای ریاضی استفاده شده است. کاربردهایی از این زبان عبارتند از:

- ریاضیات و محاسبات
- محاسبات علمی پیشرفته
- داده یابی
- مدل کردن، شبیه سازی، نمونه سازی اولیه
- آنالیز کردن اطلاعات، شناسایی و تجسم بخشیدن
- نمودارهای علمی و مهندسی
- کاربردهای پیشرفته نظیر ایجاد مبدلهای گرافیکی (GUI)

جزوه‌ای که پیش رو دارید، شامل مباحث و فرامینی است که که کاربرد Matlab را در محاسبات عددی بیان می‌کند.

این نوشتار شامل بخش‌های زیر است:

- (۱) مروری بر دستورات مقدماتی
- (۲) آشنایی با مباحث مخصوص ریاضیات تحلیلی
- (۳) آشنایی با مباحث مخصوص ریاضیات عددی

در قسمت سوم، سعی شده است که تمامی مباحث موجود در کتاب «محاسبات عددی»، نوشته «جناب آقای دکتر علی محمد پورپاک» گنجانیده شود و در پایان هر مبحث، مثالی از کتاب به همراه ذکر شماره صفحه آن آورده شده است.

برای دریافت برنامه‌هایی (M-File) هایی که در این نوشتار آورده شده است، می‌توانید از طریق e-mail با بنده مکاتبه نمایید.

mohammad.sarfaraz@gmail.com

محمد سرفراز

بهار ۱۳۸۹

MATLAB نسبت به حروف بزرگ و کوچک، حساس است. با قراردادن i در انتهای هر دستور، از چاپ نتایج جلوگیری می‌شود.

دستورات پاک کردن

clc پاک کردن صفحه نمایش

clear پاک کردن (delete) تمامی متغیرها

clear x پاک کردن (delete) متغیر x

توابع مثلثاتی

کسینوس هیپربولیک	cosh(x)	معکوس کسینوس، رادیان	acos(x)
کتانژانت، رادیان	cot(x)	معکوس کسینوس، درجه	acosd(x)
کتانژانت، درجه	cotd(x)	معکوس کسینوس هیپربولیک	acosh(x)
کتانژانت هیپربولیک	coth(x)	معکوس کتانژانت، رادیان	acot(x)
کسکانت، رادیان	csc(x)	معکوس کتانژانت، درجه	acotd(x)
کسکانت، درجه	cscd(x)	معکوس کتانژانت هیپربولیک	acoth(x)
کسکانت هیپربولیک	csch(x)	معکوس کسکانت، رادیان	acsc(x)
سکانت، رادیان	sec(x)	معکوس کسکانت، درجه	acscd(x)
سکانت، درجه	secd(x)	معکوس کسکانت هیپربولیک	acsch(x)
سکانت هیپربولیک	sech(x)	معکوس سکانت، رادیان	asec(x)
سینوس، رادیان	sin(x)	معکوس سکانت، درجه	asecd(x)
سینوس، درجه	sind(x)	معکوس سکانت هیپربولیک	asech(x)
سینوس هیپربولیک	sinh(x)	معکوس سینوس، رادیان	asin(x)
تانژانت، رادیان	tan(x)	معکوس سینوس، درجه	asind(x)
تانژانت، درجه	tand(x)	معکوس سینوس هیپربولیک	asinh(x)
تانژانت هیپربولیک	tanh(x)	معکوس تانژانت، رادیان	atan(x)
		معکوس تانژانت، درجه	atand(x)
		معکوس تانژانت هیپربولیک	atanh(x)
		کسینوس، رادیان	cos(x)
		کسینوس، درجه	cosd(x)

توابع نمایی

e^x (ورودی حقیقی و مختلط)	<code>exp(x)</code>
لگاریتم در مبنای e (ورودی حقیقی و مختلط)	<code>log(x)</code>
لگاریتم در مبنای ۲ (ورودی حقیقی و مختلط)	<code>log2(x)</code>
لگاریتم در مبنای ۱۰ (ورودی حقیقی و مختلط)	<code>log10(x)</code>
لگاریتم در مبنای e (ورودی فقط اعداد حقیقی مثبت)	<code>reallog(x)</code>
جذر (ورودی فقط اعداد حقیقی نامنفی)	<code>realsqrt(x)</code>
جذر (ورودی حقیقی و مختلط)	<code>sqrt(x)</code>
$\sqrt[y]{x}$	<code>nthroot(x,y)</code>

توابع گرد کردن

گرد کردن به سمت صفر	<code>fix(x)</code>
گرد کردن به سمت منفی بی نهایت	<code>floor(x)</code>
گرد کردن به سمت مثبت بی نهایت	<code>ceil(x)</code>
قدر مطلق	<code>abs(x)</code>
گرد کردن به سمت نزدیک ترین عدد صحیح	<code>round(x)</code>
نمایش عدد x با d رقم اعشار	<code>vpa(x,d)</code>
نمایش عدد x با d رقم اعشار	<code>maple('evalf(x,d)')</code>

توابع ریاضیات گسسته

تجزیه x به عوامل اول	<code>factor(x)</code>
$x!$	<code>factorial(x)</code>
بزرگ ترین مقسوم علیه مشترک x و y	<code>gcd(x,y)</code>
کوچک ترین مضرب مشترک x و y	<code>lcm(x,y)</code>
نمایش ۱ در صورت اول بودن x و در غیر این صورت، نمایش ۰	<code>isprime(x)</code>
$\binom{x}{y}$	<code>nchoosek(x,y)</code>
نمایش اعداد اول از ۲ تا x	<code>primes(x)</code>

توابع اعداد مختلط

i	$\sqrt{-1}$
j	$\sqrt{-1}$
$\text{abs}(z)$	محاسبه مقدار قدر مطلق
$\text{angle}(z)$	محاسبه مقدار زاویه بر حسب رادیان
$\text{conj}(z)$	محاسبه مزدوج
$\text{imag}(z)$	نمایش قسمت موهومی
$\text{real}(z)$	نمایش قسمت حقیقی
$\text{isreal}(z)$	اگر z عدد حقیقی باشد، مقدار ۱ و اگر متلط باشد، صفر را برمی گرداند.
$\text{complex}(a,b)$	ایجاد یک عدد مختلط به فرم $a+bi$

(مثال)

>> z1=2+3*i

z1 =

2.0000 + 3.0000i

>> z2=-5+j

z2 =

-5.0000 + 1.0000i

>> r=abs(z1)

r =

3.6056

>> a=angle(z2)

a =

2.9442

```
>> z3=conj(z1)

z3 =

    2.0000 - 3.0000i

>> a1=real(z1)

a1 =

    2

>> b1=imag(z1)

b1 =

    3

>> isreal(z2)

ans =

    0

>> z3=complex(7,-8)

z3 =

    7.0000 - 8.0000i

>> z4=z1*z2-z3

z4 =

   -20.0000 - 5.0000i
```

فرمت‌ها

format long	نمایش خروجی با ۱۴ یا ۱۵ رقم اعشار
format short	نمایش خروجی با ۴ رقم اعشار
format short e	نمایش خروجی با ۴ رقم اعشار و نماد علمی
format long e	نمایش خروجی با ۱۴ یا ۱۵ رقم اعشار و نماد علمی
format rat	نمایش خروجی به صورت کسری

Ex.) $\sqrt{\sinh(0.125)} + \cos(35^\circ) * \sqrt[6]{\ln 25} + \left(\frac{10}{3}\right) * [\tanh 17] + \log_7 23 = ?$

```

Command Window
>> clear
>> a=sqrt(sinh(0.125))+cosd(35)+nthroot(log(25),6)+nchoosek(10,3)*floor(tanh(17))+(log(23))/log(7)

a =

    3.999606098686637

>> vpa(a,10)

ans =

    3.999606099

>> format long
>> a

a =

    3.999606098686637

>> vpa(a,11)

ans =

    3.9996060987

>> disp(a)
    3.999606098686637

>> display(a)

a =

    3.999606098686637

>> format rat
>> a

a =

    10155/2539

```

معرفی تابع یک متغیره

مثال) می‌خواهیم تابع $y=f(x)=1+\sin^2(x)$ را معرفی، مقدار دهی و رسم کنیم.

```
>> clear
>> format long
>> syms x
>> f=inline('1+(sin(x))^2','x')

f =

    Inline function:
    f(x) = 1+(sin(x))^2

>> f(0)

ans =

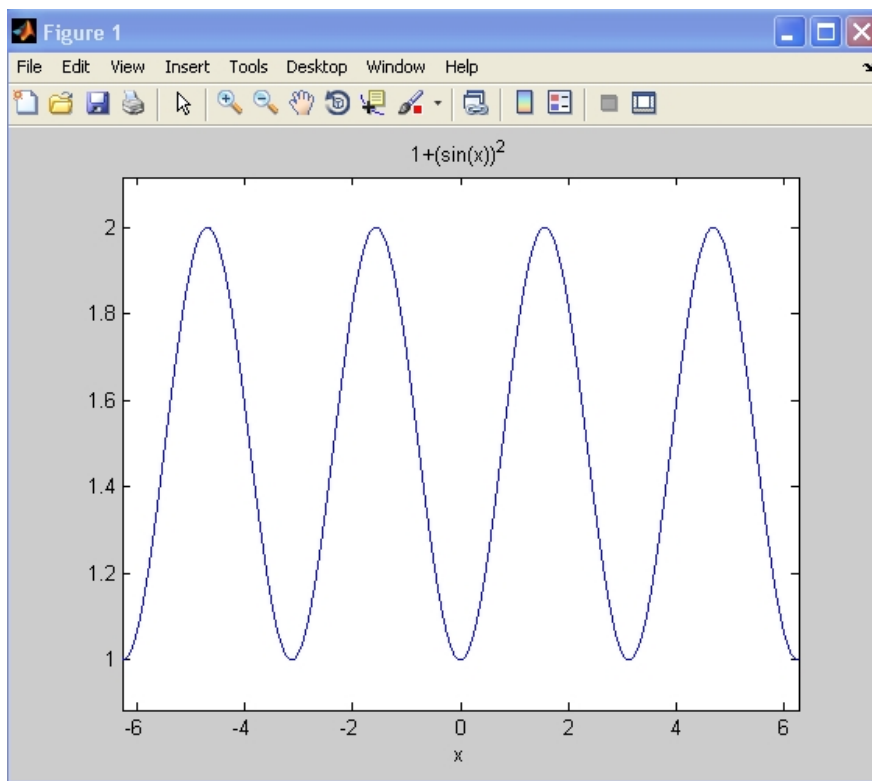
    1

>> f(pi+atan(0.25))

ans =

    1.058823529411765

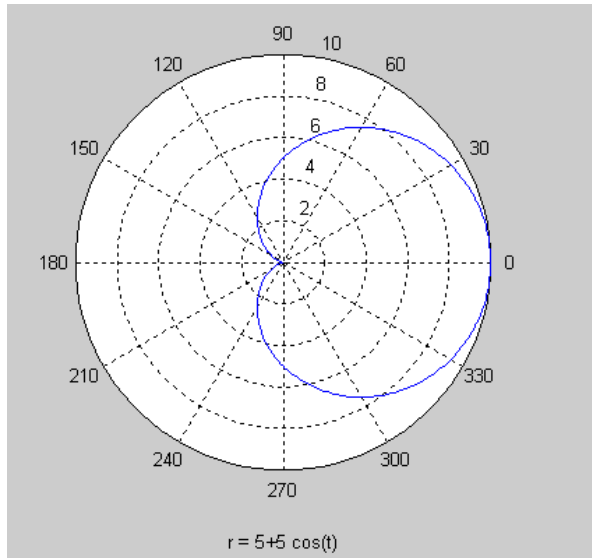
>> ezplot(f)
>>
```



رسم در مختصات قطبی

مثال) مطلوبست ترسیم تابع $r=5(1+\cos \theta)$

```
>> syms t
>> ezpolar(5*(1+cos(t)))
```



رسم توابع یک متغیره با ezplot

مثال) مطلوبست نمایش تابع $y=x+\sinh(x)$

```
>> syms x
>> ezplot(x+sinh(x))
>>
```

مثال) مطلوبست نمایش تابع $y=x+\sinh(x)$ در فاصله $-2\pi \leq x \leq 2\pi$.

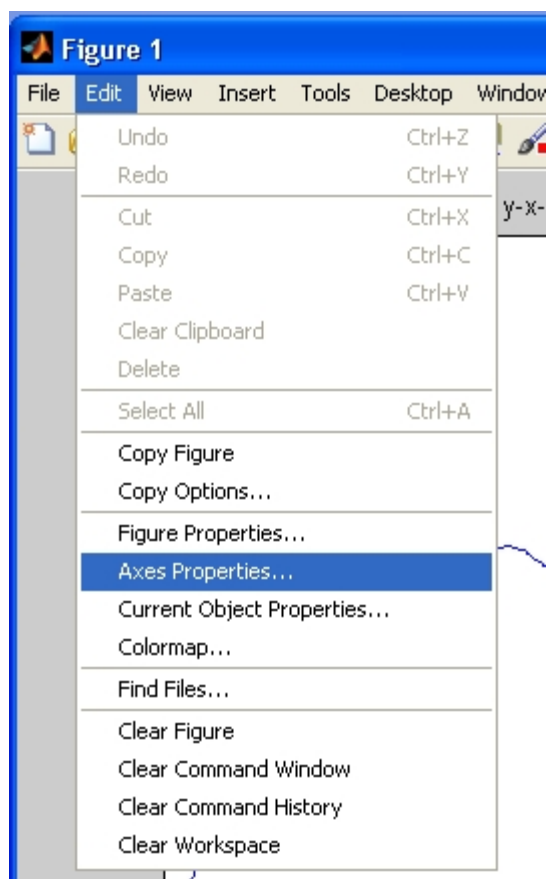
```
>> syms x
>> ezplot(x+sinh(x), [-2*pi 2*pi])
>>
```

مثال) مطلوبست نمایش تابع $y=x+\sinh(x)$ و تابع $y=x+\sin^2(x)$ در کنار هم.

```
>> syms x
>> ezplot(x+sinh(x))
>> hold on
>> ezplot(x+(sin(x))^2)
>>
```

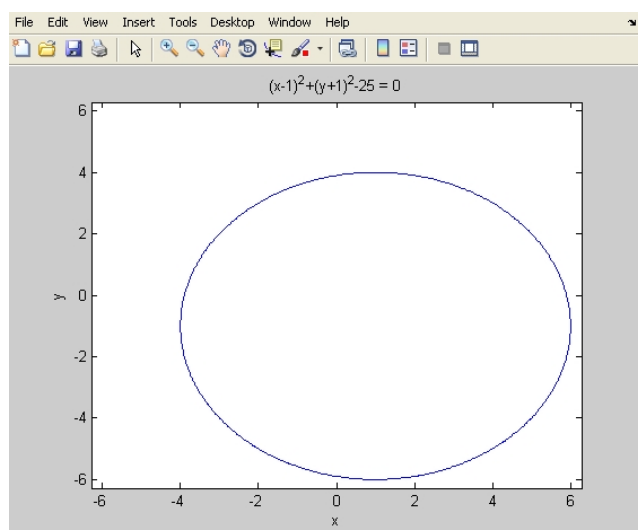
تذکر: برای تغییر مشخصات گراف مانند حدود x و y ، تغییر رنگ، لگاریتمی کردن مقیاس x یا y و... از منوی

Edit گزینه Axes Properties... را انتخاب می کنیم.



مثال) نمودار دایره $(x-1)^2 + (y+1)^2 = 25$ را رسم کنید.

```
>> syms x y
>> ezplot(' (x-1)^2+(y+1)^2-25' )
>>
```



به دلیل اینکه مقیاس محورها برابر نیست، نمودار دایره شبیه بیضی شده است. کافی است دستور زیر را اجرا کنیم.

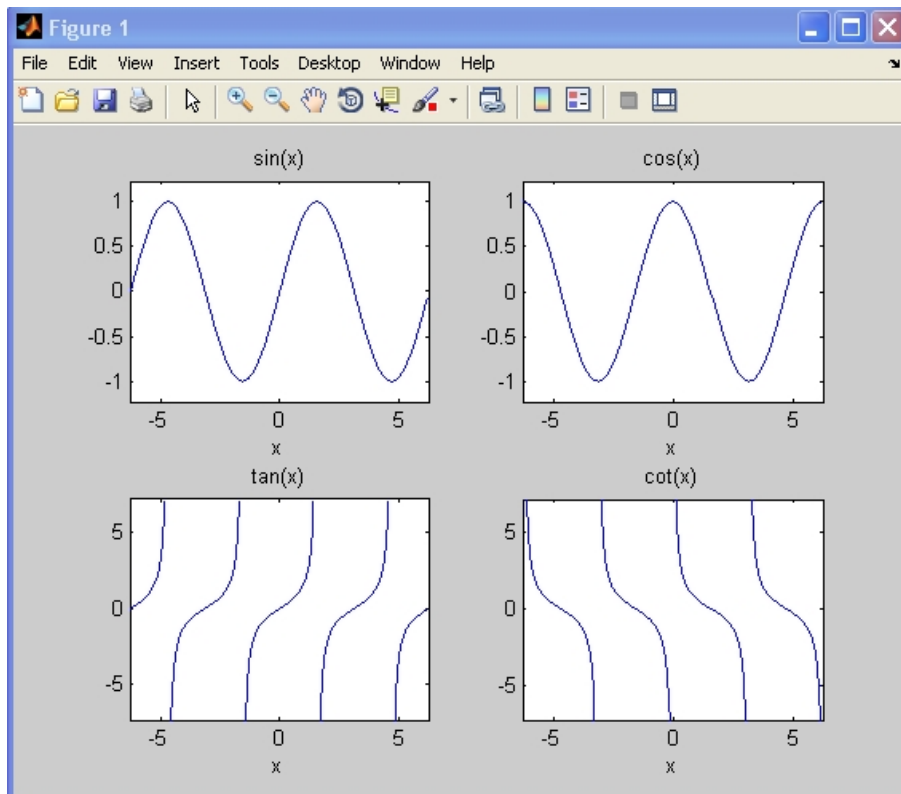
```
>> axis equal
>>
```

مثال) مطلوبست رسم نمودار رابطه $\sqrt{x^2 + y^2} \times (x^2 - y^2) = 3xy$

```
>> ezplot('sqrt(x^2+y^2)*(x^2-y^2)-3*x*y')
>>
```

مثال) می‌خواهیم نمودارهای $y=\sin(x)$ و $y=\cos(x)$ و $y=\tan(x)$ و $y=\cot(x)$ را در یک پنجره گراف و جدا از هم رسم کنیم.

```
>> syms x
>> subplot(2,2,1)
>> ezplot(sin(x))
>> subplot(2,2,2)
>> ezplot(cos(x))
>> subplot(2,2,3)
>> ezplot(tan(x))
>> subplot(2,2,4)
>> ezplot(cot(x))
```



معرفی تابع دو متغیره

مثال) می‌خواهیم تابع $f(x,y)=1+xy-\sin(x+2y)$ را معرفی و مقداردهی کنیم.

```
>> syms x y
>> f=inline('1+x*y-sin(x+2*y)','x','y')
```

```
f =
```

```
Inline function:
f(x,y) = 1+x*y-sin(x+2*y)
```

```
>> a=f(pi,2)
```

```
a =
```

```
6.526382811871658
```

مقداردهی با دستور subs

(مثال)

```
>> syms a b x c
>> f=a*x^2+b*x+c
```

```
f =
```

```
a*x^2+b*x+c
```

```
>> m=subs(f,x,2)
```

```
m =
```

```
4*a+2*b+c
```

در مثال بالا مقدار ۲ در متغیر x جایگذاری شده است.

(مثال)

```
>> syms a b x c
>> f=a*x^2+b*x+c
```

```
f =
```

```
a*x^2+b*x+c
```

```
>> k=subs(f,{x,a,b,c},{2,-3,4,8})
```

$k =$

4

در مثال بالا مقدار ۲ در متغیر x ، مقدار ۳- در متغیر a ، مقدار ۴ در متغیر b ، مقدار ۸ در متغیر c ، جایگذاری شده است.

عملیات بر روی چند جمله‌ای‌ها (مثال)

```
>> syms t x y
>> f=(x+2)^3+4*y*t+y*x+t*x+4*y*(x+2);
>> g=(x^2-1)*(x-2)*(x-3);
>> collect(f,x)
```

ans =

$x^3+6x^2+(12+t+5y)x+8+8y+4yt$

```
>> collect(g,x)
```

ans =

$-6+x^4-5x^3+5x^2+5x$

```
>> e=x^4+4;
>> factor(e)
```

ans =

$(x^2-2x+2)*(x^2+2x+2)$

```
>> factor(g)
```

ans =

$(x-1)*(x+1)*(x-2)*(x-3)$

```

>> h=cos(x+y);
>> expand(g)

ans =

-6+x^4-5*x^3+5*x^2+5*x

>> expand(h)

ans =

cos(x)*cos(y)-sin(x)*sin(y)

>> expand(f)

ans =

x^3+6*x^2+12*x+8+4*y*t+5*y*x+t*x+8*y
>> p=(cos(3*x))^2+(sin(3*x))^2

p =

cos(3*x)^2+sin(3*x)^2

>> simplify(p)

ans =

1
> b=(1/x^3+6/x^2+12/x+8)^(1/3);
>> b1=simple(b);
>> b1

b1 =

(2*x+1)/x

>> b2=simplify(b);
>> b2

```

b2 =

 $((2*x+1)^3/x^3)^{(1/3)}$

همانطور که ملاحظه می‌گردد، دستور collect، f را بر حسب x مرتب می‌کند. دستور factor، e را تجزیه می‌کند. دستور expand بسط می‌دهد و دستور simple و simplify عبارت را ساده می‌کند.

نمایش با دستور pretty

```
>> syms s
>> n=(10*s^2+40*s+30)/(s^2+6*s+8);
>> pretty(n)
```

$$\frac{10s^2 + 40s + 30}{s^2 + 6s + 8}$$

>>

محاسبه حدود توابع

$$Ex.) \lim_{x \rightarrow 0} \frac{1 - \cos x}{x^2} = ?$$

```
>> syms x
>> l=limit((1-cos(x))/(x^2),x,0)
```

l =

1/2

$$Ex.) \lim_{x \rightarrow +\infty} \frac{\sqrt{ax^2 + 2x}}{2x + \sin(x)} = ?$$

```
>> syms x a
>> l=limit((sqrt(a*x^2+2*x))/(2*x+sin(x)),x,+inf)
```

l =

 $1/2*a^{(1/2)}$

$$Ex.) \lim_{x \rightarrow \pi} [\sin x] = ?$$

```
>> syms x
>> l=limit(floor(sin(x)),x,pi,'left')
```

l =

0

Ex.) $\lim_{x \rightarrow \pi^+} [\sin x] = ?$

```
>> l=limit(floor(sin(x)),x,pi,'right')
```

l =

-1

Ex.) $\lim_{x \rightarrow \pi} [\sin x] = ?$

```
>> l=limit(floor(sin(x)),x,pi)
```

l =

NaN

دقت شود که منظور از NaN، Not a Number می باشد.

محاسبه مشتق توابع

Ex.) $\frac{d}{dx} (x + \tan x) = ?$

```
>> syms x
```

```
>> f=x+tan(x);
```

```
>> d=diff(f,x)
```

d =

2+tan(x)^2

Ex.) $\frac{d^4}{dx^4} (x + \tan x) = ?$

```
>> syms x
```

```
>> f=x+tan(x);
```

```
>> d4=diff(f,x,4)
```

d4 =

16*(1+tan(x)^2)^2*tan(x)+8*tan(x)^3*(1+tan(x)^2)

اگر می‌خواستیم مقدار مشتق چهارم f را در نقطه $x = -7.5$ حساب کنیم:

```
>> syms x
>> f=x+tan(x);
>> d4=diff(f,x,4);
>> subs(d4,x,-7.5)
```

ans =

-4.318149972714537e+003

Ex.) $\frac{\partial^2 f}{\partial y^2} (y^3 x^2 + y^2 x + z^2 x y^4) = ?$

```
>> syms x y z
>> f=y^3*x^2+y^2*x+z^2*x*y^4;
>> d2=diff(f,y,2)
```

d2 =

6*y*x^2+2*x+12*z^2*x*y^2

Ex.) $\frac{\partial^2 f}{\partial x \partial y} (y^3 x^2 + y^2 x + z^2 x y^4) = ?$

```
>> syms x y z
>> f=y^3*x^2+y^2*x+z^2*x*y^4;
>> d=diff(diff(f,y),x)
```

d =

6*y^2*x+2*y+4*z^2*y^3

محاسبه انتگرال معین

Ex.) $\int e^{ax} \cos(bx) dx = ?$

```
>> syms a b x
>> k=int(exp(a*x)*cos(b*x),x)
```

k =

$a/(a^2+b^2)*\exp(a*x)*\cos(b*x)+b/(a^2+b^2)*\exp(a*x)*\sin(b*x)$

```
>> pretty(k)
```

$$\frac{a \exp(a x) \cos(b x)}{a^2 + b^2} + \frac{b \exp(a x) \sin(b x)}{a^2 + b^2}$$

```
>>
```

محاسبه انتگرال نامعین

Ex.) $\int_{-5}^{7.6} x e^{ax} dx = ?$

```
>> syms a x
```

```
>> k=int(x*exp(a*x),x,-5,7.6)
```

```
k =
```

$$\frac{1}{5} (5 \exp(-5a) + 25 \exp(-5a)a - 5 \exp(38/5 a) + 38 \exp(38/5 a)a) / a^2$$

```
>> pretty(k)
```

$$\frac{5 \exp(-5 a) + 25 \exp(-5 a) a - 5 \exp(38/5 a) + 38 \exp(38/5 a) a}{1/5 a^2}$$

```
>> collect(k,a)
```

```
ans =
```

$$(5 \exp(-5a) + 38/5 \exp(38/5 a)) / a + (\exp(-5a) - \exp(38/5 a)) / a^2$$

```
>> pretty(ans)
```

$$\frac{5 \exp(-5 a) + 38/5 \exp(38/5 a)}{a} + \frac{\exp(-5 a) - \exp(38/5 a)}{a^2}$$

```
>>
```

$$Ex.) \int_0^{+\infty} \frac{1}{1+x^2} dx = ?$$

```
>> syms x
```

```
>> h=int(1/(1+x^2),x,0,+inf)
```

```
h =
```

```
1/2*pi
```

محاسبه انتگرال دوگانه

$$Ex.) \int_1^{e^4} \int_0^x \frac{dy dx}{(x+y)^2} = ?$$

```
>> syms x y
```

```
>> b=int(int(1/(x+y)^2,y,0,x),x,1,exp(4))
```

```
b =
```

```
1/2*log(960500813064011)-22*log(2)
```

```
>> vpa(b,3)
```

```
ans =
```

```
2.0
```

تذکر: در مبحث ریاضیات عددی، توابع `dblquad` و `triplequad` برای محاسبه انتگرال دوگانه و سه گانه به روش عددی به کار می‌روند.

محاسبه سری

$$Ex.) s_1 = \sum_{k=0}^5 k^2 = ?$$

```
>> syms k
>> s1=symsum(k^2,k,0,5)
```

s1 =

55

$$Ex.) s_2 = \sum_{k=n}^p k^2 = ?$$

```
>> syms k n p
>> s2=symsum(k^2,k,n,p)
```

s2 =

$1/3*(p+1)^3 - 1/2*(p+1)^2 + 1/6*p + 1/6 - 1/3*n^3 + 1/2*n^2 - 1/6*n$

```
>> pretty(s2)
```

$$\frac{1}{3} (p + 1)^3 - \frac{1}{2} (p + 1)^2 + \frac{1}{6} p + \frac{1}{6} - \frac{1}{3} n^3 + \frac{1}{2} n^2 - \frac{1}{6} n$$

```
>> pretty(collect(s2))
```

$$\frac{1}{3} p^3 + \frac{1}{2} p^2 + \frac{1}{6} p - \frac{1}{3} n^3 + \frac{1}{2} n^2 - \frac{1}{6} n$$

```
>>
```

$$Ex.) s_3 = \sum_{k=0}^{+\infty} \frac{x^k}{k!} = ?$$

```
>> syms x k
>> s3=symsum((x^k)/(sym('k!')),k,0,+inf)
```

s3 =

exp(x)

محاسبه سری تیلور توابع یک متغیره

مثال) مطلوبست سری تیلور تابع زیر در $a=8$.

$$Ex.) f(x) = \frac{1}{5 + 4 \cos x}$$

```
>> syms x
>> f=1/(5+4*cos(x));
>> t=taylor(f,8)
```

t =

$$1/9 + 2/81 x^2 + 5/1458 x^4 + 49/131220 x^6$$

```
>> pretty(t)
```

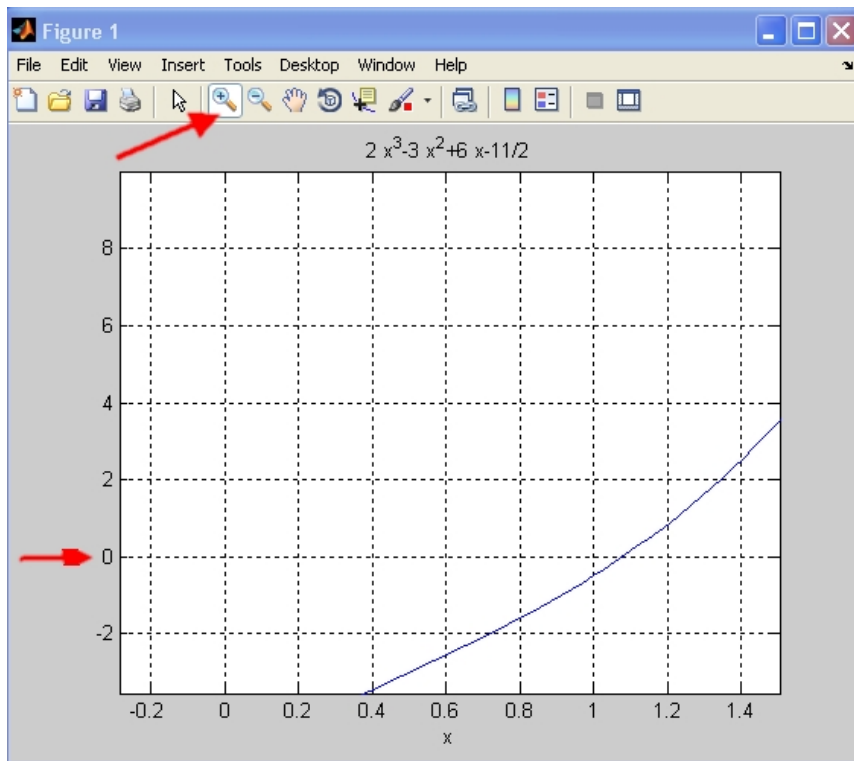
$$1/9 + 2/81 x^2 + 5/1458 x^4 + \frac{49}{131220} x^6$$

پیدا کردن ریشه‌های معادله $f(x)=0$

$$Ex.) 2x^3 - 3x^2 + 6x - 5.5 = 0$$

برای حدس اولیه ابتدا نمودار تابع را رسم می‌کنیم و با ابزار zoom می‌توان محل ریشه (ها) را یافت.

```
>> syms x
>> ezplot(2*x^3-3*x^2+6*x-5.5)
>> grid on
```



```
>> [x]=solve('2*x^3-3*x^2+6*x-5.5','x')
```

```
x =
```

```
1.0799656351083685970101250717027
```

```
.21001718244581570149493746414863+1.58185653693057450159354822  
31862*i
```

```
.21001718244581570149493746414863-  
1.5818565369305745015935482231862*i
```

```
>> vpa(x,4)
```

```
ans =
```

```
1.080  
.2100+1.582*i  
.2100-1.582*i
```

Ex.) $x^3 - 6.7 * x^2 + 13.86 * x = 8.712$

```
>> syms x
```

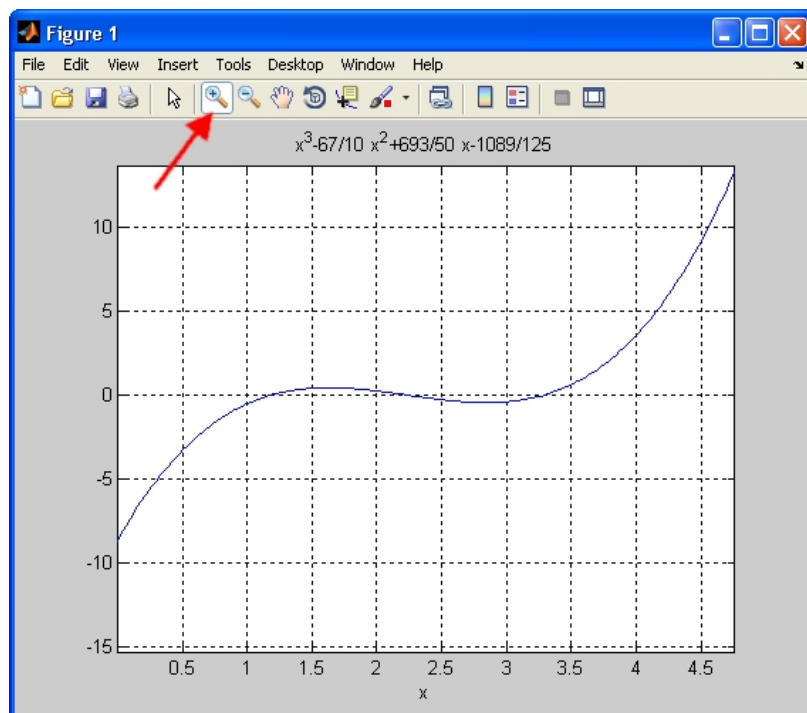
```
>> ezplot(x^3-6.7*x^2+13.86*x-8.712)
```

```
>> grid
```

```
>> [m]=solve('x^3-6.7*x^2+13.86*x-8.712','x')
```

m =

```
1.20000000000000000000000000000000
2.20000000000000000000000000000000
3.30000000000000000000000000000000
```



Ex.) $ax^2 + bx + c = 0$

```
>> syms a x b c
>> [m]=solve('a*x^2+b*x+c','x')
```

m =

```
-1/2*(b-(b^2-4*c*a)^(1/2))/a
-1/2*(b+(b^2-4*c*a)^(1/2))/a
```

```
>> pretty(m)
```

```

      2      1/2
      b - (b - 4 c a)
[- 1/2 -----]
      a
[
      2      1/2
      b + (b - 4 c a)
[- 1/2 -----]
      a
]
```

```
>>
```

نکته بسیار مهم: اگر در حل معادله $f(x)=0$ بیش از یک ریشه داشته باشیم و دنبال ریشه‌ای باشیم که حدود آنرا می‌دانیم از روش‌های زیر استفاده می‌کنیم.

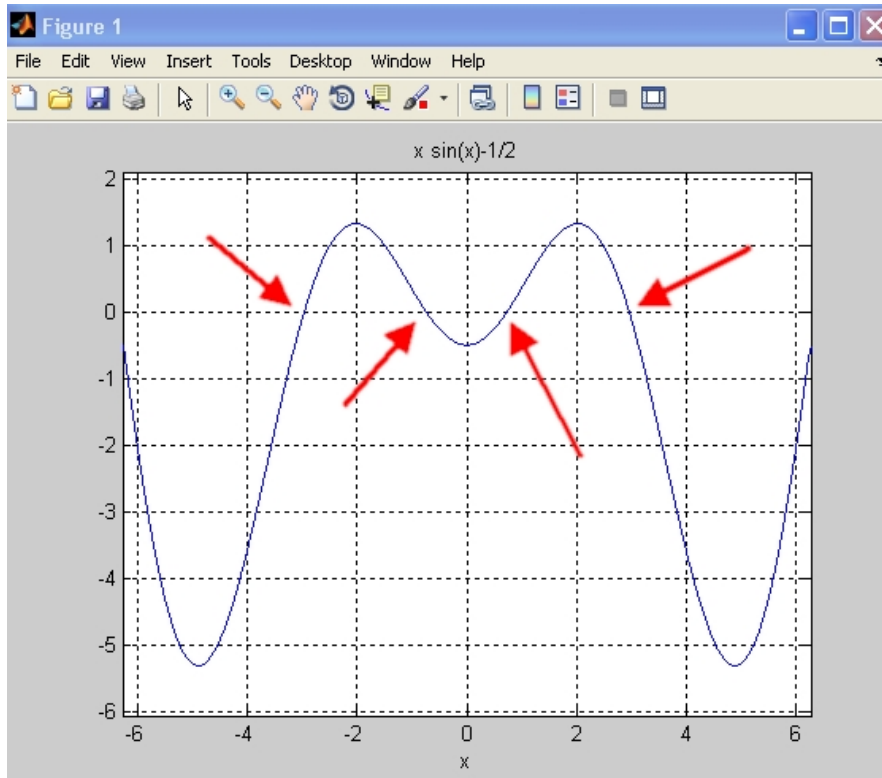
مثال) مطلوبست ریشه معادله $x\sin(x)=0.5$. ریشه بین ۲ و ۳ است.

اگر مانند روش قبل حل کنیم:

```
>> syms x
>> ezplot(x*sin(x)-0.5)
>> grid
>> [m]=solve('x*sin(x)-0.5','x')
```

m =

```
- .74084095509549062101093540994313
```



مشخص است که Matlab جواب مورد نظر را نداده است. برای یافتن جواب به روش زیر عمل می‌کنیم:

```
>> a=maple('fsolve(x*sin(x)-0.5=0,x,2..3)')
```

a =

```
2.9725854903823601148057972025655
```

روش دوم)

```
>> f=inline('x*sin(x)-0.5','x')
```

f =

Inline function:

f(x) = x*sin(x)-0.5

```
>> m=fzero(f,[2 3])
```

m =

```
2.9726
```

```
>> format long
```

```
>> m
```

m =

2.972585490382360

حل دستگاه معادلات

$$Ex.) \begin{cases} x + y + 2z = -3 \\ 2x + 2y + z = 3 \\ -x + 2y + z = 0 \end{cases}$$

>> syms x y z

>> [x y z]=solve('x+2*y+2*z+3','2*x+2*y+z-3','-x+2*y+z','x','y','z')

x =

1

y =

3

z =

-5

نکته بسیار مهم: نام گذاری خروجی این دستور به ترتیب حروف الفباست. به دستورات زیر دقت کنید.

```
>> syms x y z
>> [x y z]=solve('x+2*y+2*z+3','2*x+2*y+z-3','-x+2*y+z','x','z','y')

x =

1

y =

3

z =

-5

>> syms x y z
>> [a b c]=solve('x+2*y+2*z+3','2*x+2*y+z-3','-x+2*y+z','z','x','y')

a =

1

b =

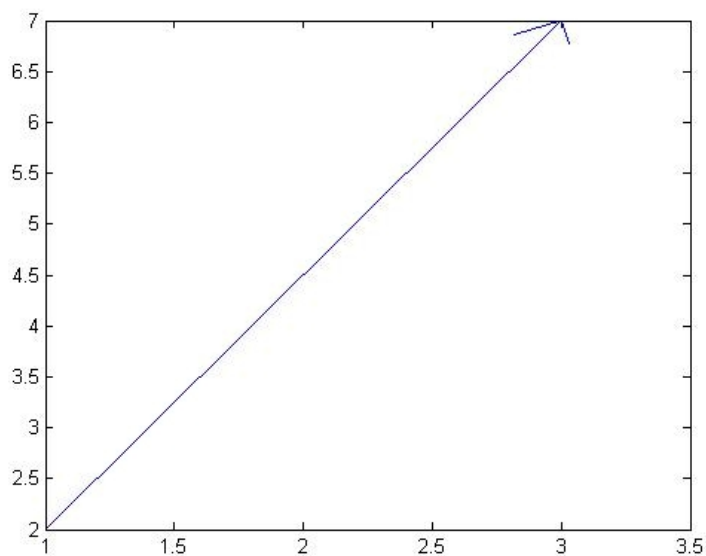
3

c =

-5
```

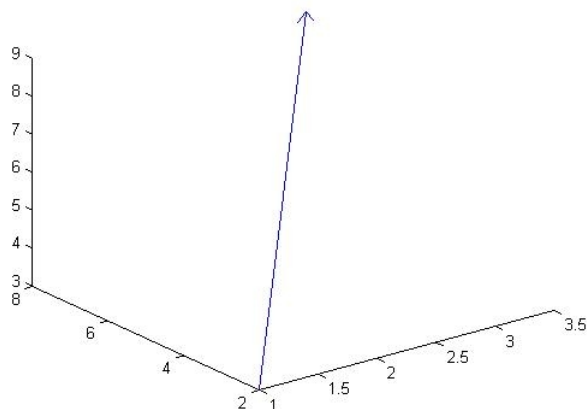
رسم بردار در فضای دو بعدی

```
>> A=[1 2];
>> B=[2 5];
>> arrow(A,B)
```



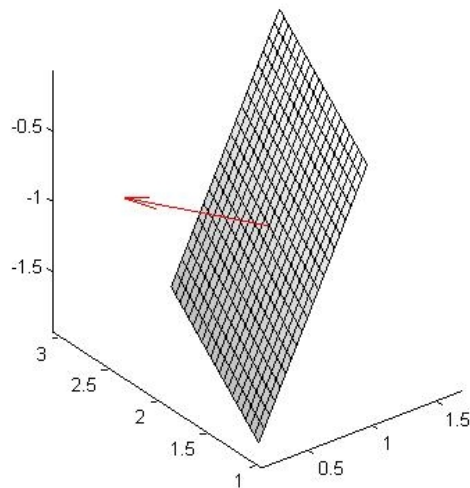
رسم بردار در فضای سه بعدی

```
>> A=[1 2 3];
>> B=[2 5 6];
>> arrow3(A,B)
```



رسم صفحه

```
>> P0=[1 2 -1];
>> n=[-5 2 2];
>> plane(P0,n)
```



مشخص کردن بردارهای کنج فرنه (T&N&B)

```
>> x=inline('2*t');
>> y=inline('t.^2');
>> z=inline('t.^3/2');
>> t=0:.01:12;
>> plot3(x(t),y(t),z(t));
>> axis equal
>> hold on
>> frenet(x,y,z)
enter a value of t 0
```

t =

0

T	N	B
---	---	---

frame =

1.0000	0	-0.0000
0	1.0000	0
0.0000	0	1.0000

kappa	a_T	a_N
-------	-----	-----

ans =

```

1.4142      0      2.0000

enter a value of t      0.8

t =

0.8000

T      N      B

frame =

0.7312    -0.6157    0.2937
0.5850     0.3444   -0.7343
0.3510     0.7088    0.6119

kappa      a_T      a_N

ans =

1.4449     2.0122     2.3897

enter a value of t      1.4

t =

1.4000

T      N      B

frame =

0.4419    -0.7206    0.5343
0.6187    -0.1865   -0.7632
0.6496     0.6678    0.3634

kappa      a_T      a_N

ans =

1.1432     3.9657     2.4320

```

```
enter a value of t 1.8
```

```
t =
```

```
1.8000
```

```
T          N          B
```

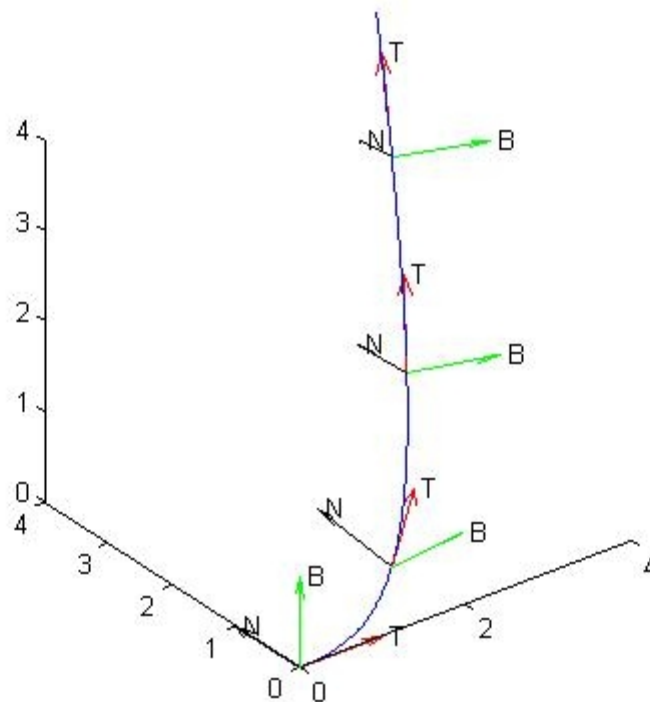
```
frame =
```

```
0.3140    -0.6967    0.6450
0.5651    -0.4088   -0.7166
0.7629     0.5895    0.2654
```

```
kappa      a_T      a_N
```

```
ans =
```

```
0.9373    5.2497    2.3657
```



ترسیم توابع عددی دو متغیره

تابع $z = f(x, y) = 10x^2 + y^2$ را رسم می کنیم.

```
>> x=-1:.1:1;
```

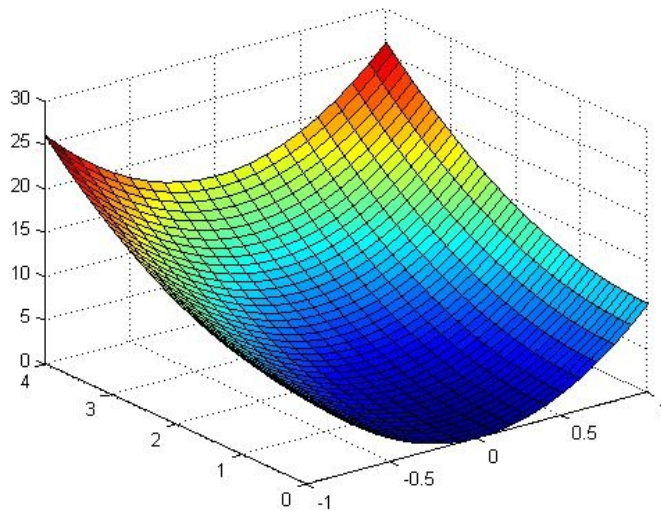
```
>> y=0:.1:4;
```

```
>> [X,Y]=meshgrid(x,y);
>> f=inline('10*x.^2+y.^2','x','y')
```

f =

Inline function:
f(x,y) = 10*x.^2+y.^2

```
>> surf(X,Y,f(X,Y))
```



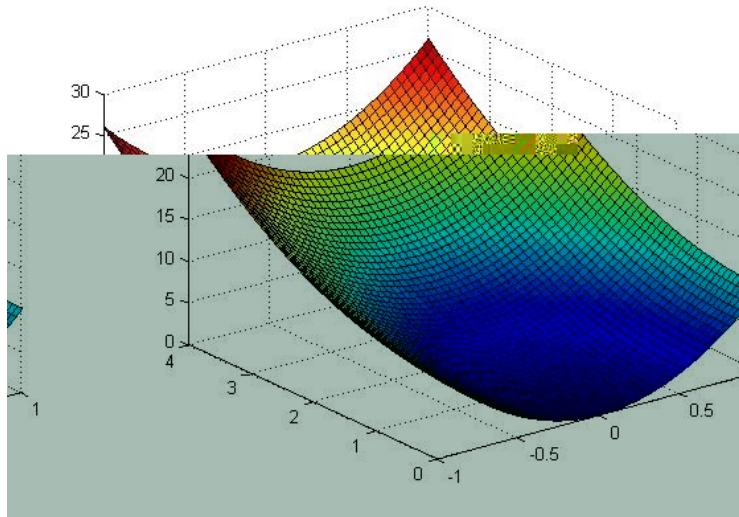
- راه کوتاه: ابتدا محدوده صفحه X-Y را در corners وارد می کنیم.

```
>> clear
>> f=inline('10*x.^2+y.^2','x','y')
```

f =

Inline function:
f(x,y) = 10*x.^2+y.^2

```
>> corners=[-1 1 0 4];
>> qsurf(f,corners)
```

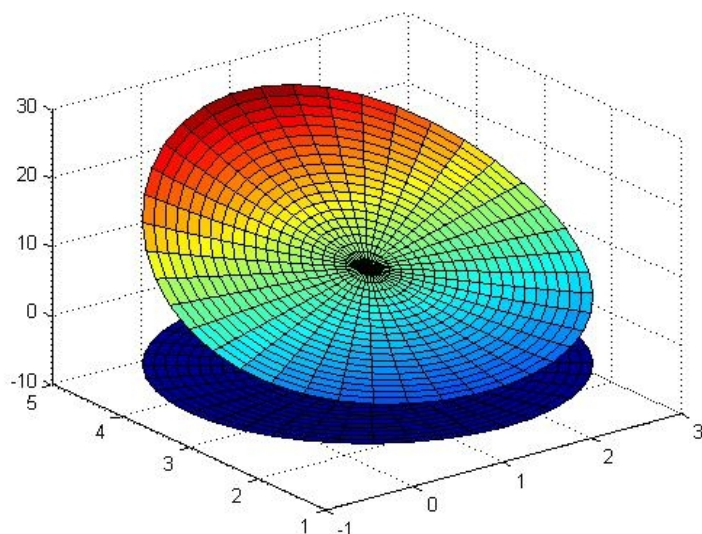


ملاحظه می گردد که mesh بندی دقیقتر است.

ترسیم توابع دو متغیره با مختصات قطبی

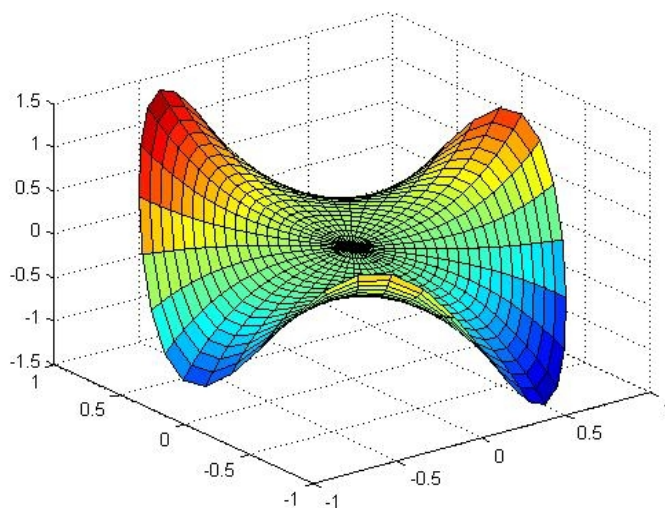
- رسم $f(x, y) = x - 1 + y^2$ بر روی ناحیه مدور $\{(x-1)^2 + (y-3)^2 = 4\}$

```
>> %first make a meshgrid in r,theta-coordinates
>> r=linspace(0,2,21);
>> theta=linspace(0,2*pi,41);
>> [R,TH]=meshgrid(r,theta);
>> %now convert into a curvilinear
>> X=1+R.*cos(TH);
>> Y=3+R.*sin(TH);
>> Z=X-1+Y.^2;
>> surf(X,Y,Z)
>> %add the plane Z=-5
>> hold on
>> surf(X,Y,-5+0*Z)
>> hold off
```



- رسم $f(r, q) = \frac{r}{2} \sin(q) + r^3 \cos(3q)$ بر روی ناحیه دایره ای که مرکزش در مبدا است.

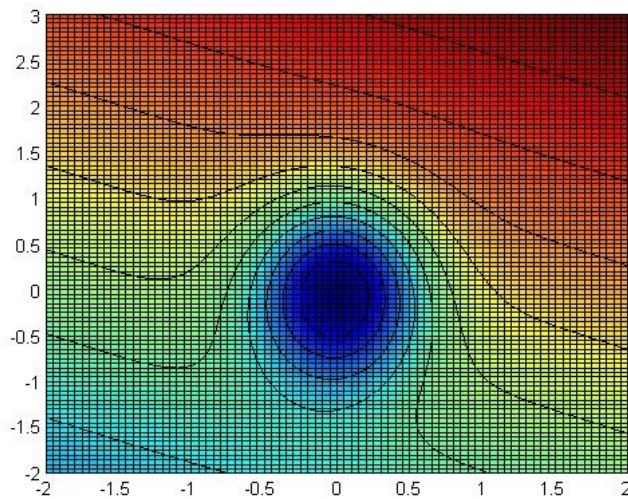
```
>> r=linspace(0,1,21);
>> theta=linspace(0,2*pi,41);
>> [R,TH]=meshgrid(r,theta);
>> X=R.*cos(TH);
>> Y=R.*sin(TH);
>> Z=0.5*R.*sin(TH)+R.^3.*cos(3*TH);
>> surf(X,Y,Z)
```



منحنیهای تراز

- تابع $f(x, y) = 6e^{-3x^2 - y^2} + \frac{x}{2} + y$ را در نظر بگیرید. ۱۰ خط تراز به رنگ سیاه در ناحیه $[-2, 2] \times [-2, 3]$ رسم می کنیم.

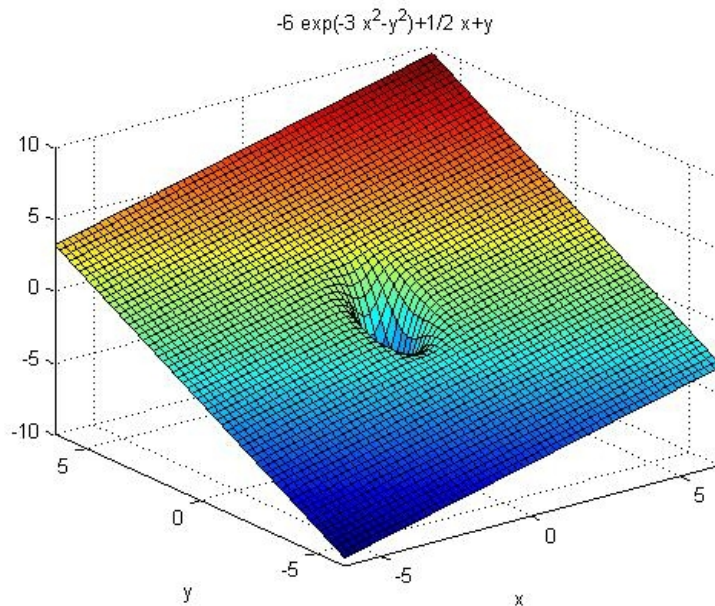
```
>> x=-2:.05:2;
>> y=-2:.05:3;
>> [X,Y]=meshgrid(x,y);
>> f=inline('-6*exp(-3*x.^2-y.^2)+.5*x+y','x','y');
>> Z=f(X,Y);
>> pcolor(X,Y,Z)
>> hold on
>> contour(X,Y,Z,10,'k')
>> hold off
```



تکنیک رسم سریع توابع دو متغیره

تابع بالا را رسم می کنیم:

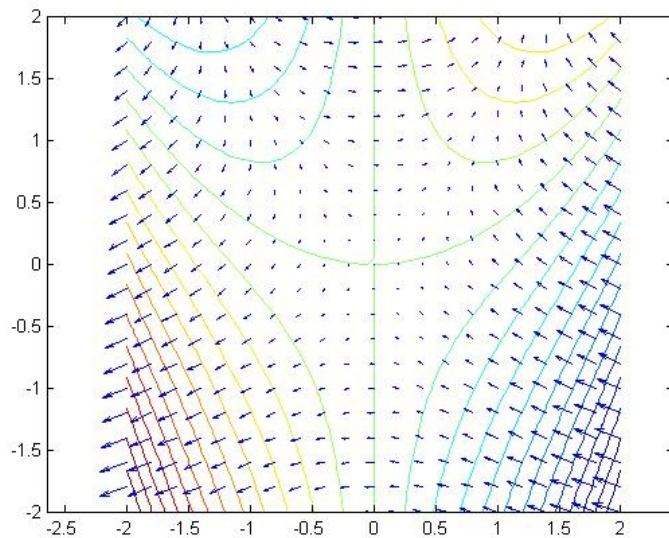
```
>> syms x y
>> f=-6*exp(-3*x^2-y^2)+.5*x+y;
>> ezsurf(f)
```



رسم بردار گرادیان و منحنی تراز

- تابع $f(x,y) = xy - \frac{x^3}{3}$ را در نظر بگیرید. بردار گرادیان و منحنی تراز را در ناحیه $[-2, 2] \times [-2, 2]$ رسم کنید.

```
>> f=inline('x.*y-(x.^3)/3','x','y');
>> fx=inline('y-x.^2','x','y');
>> fy=inline('x','x','y');
>> x=-2:0.05:2;
>> y=x;
>> [X,Y]=meshgrid(x,y);
>> Z=f(X,Y);
>> levels=[-6:0.5:6];
>> contour(X,Y,Z,levels)
>> hold on
>> xx=-2:0.2:2;
>> yy=xx;
>> [XX,YY]=meshgrid(xx,yy);
>> U=fx(XX,YY);
>> V=fy(XX,YY);
>> quiver(XX,YY,U,V)
>> axis equal
```



رسم سطوح تراز توابع سه متغیره

همان طور که می دانیم معادله سطوح تراز به صورت $f(x,y,z)=c$ می باشد. این ترسیم به کمک M-File ای

به صورت `impl(f, corners, c)` انجام می شود.

- تابع $f(x,y,z) = x^2 + y^2 - z^2$ و برای $c=0.1, 1, 0, -0.5$

```
>> f=inline('x.^2+y.^2-z.^2','x','y','z');
>> corners=[-4 4 -4 4 -4 4];
>> subplot(2,2,1)
>> impl(f,corners,0)
```

ans =

The max over this domain is 32.00000

ans =

The min over this domain is -16.00000

```
>> subplot(2,2,2)
>> impl(f,corners,0.1)
```

ans =

The max over this domain is 32.00000

ans =

The min over this domain is -16.00000

```
>> subplot(2,2,3)
```

```
>> impl(f,cornerRadius,-0.5)
```

ans =

The max over this domain is 32.00000

ans =

The min over this domain is -16.00000

```
>> subplot(2,2,4)
```

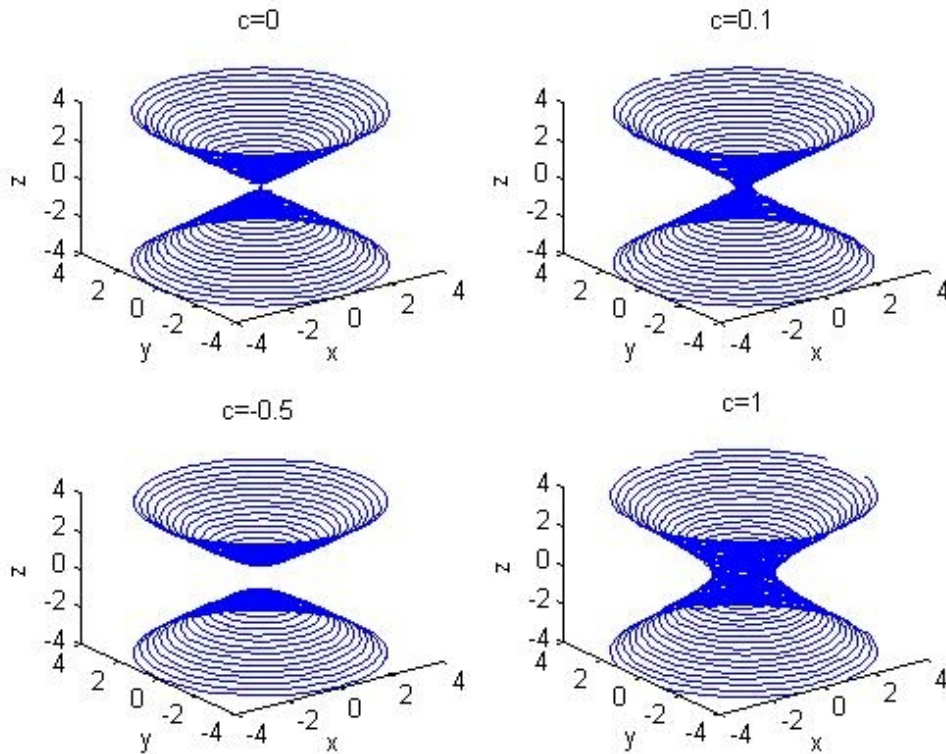
```
>> impl(f,cornerRadius,1)
```

ans =

The max over this domain is 32.00000

ans =

The min over this domain is -16.00000

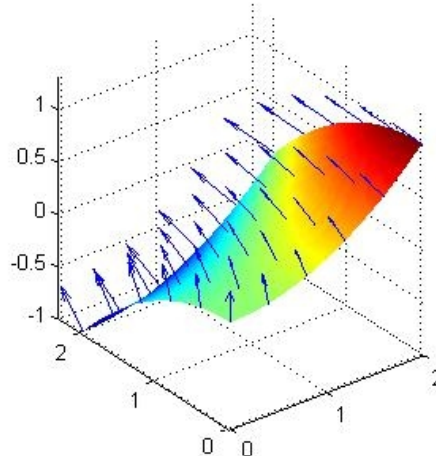
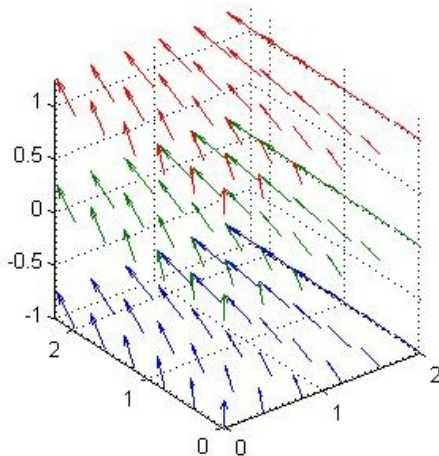


ترسیم میدان برداری گرادیان

تابع $f(x, y, z) = z + \frac{(y^2 - x^2)}{4}$ را در نظر بگیرید. میدان برداری گرادیان آن به صورت $\nabla f = (-\frac{x}{2}, \frac{y}{2}, 1)$ می باشد.

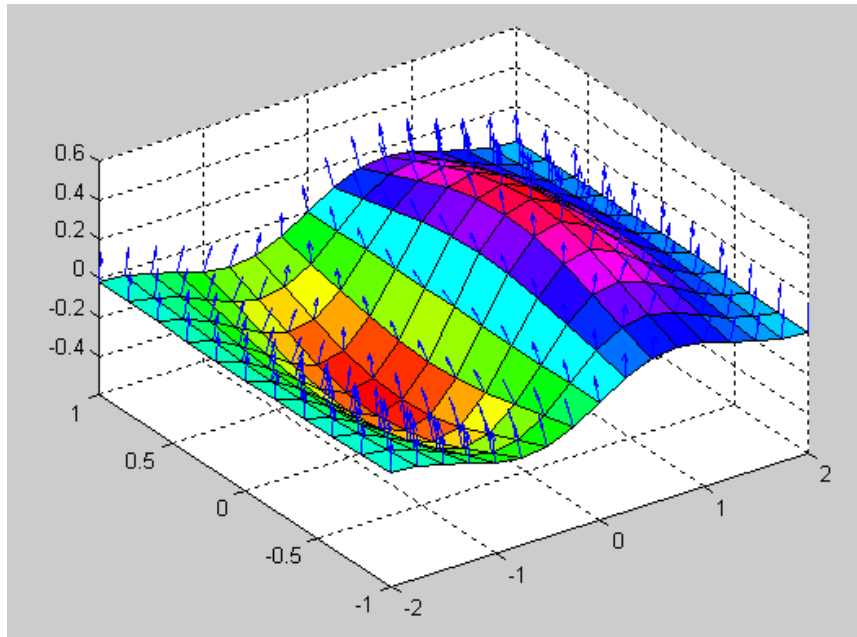
```
>> [X,Y]=meshgrid(0:0.4:2);
>> U=-X/2;
>> V=Y/2;
>> W=1+0*X;
>> subplot(1,2,1)
>> for z=[-1,0,1]
Z=z+0*X;
quiver3(X,Y,Z,U,V,W)
hold on
end
>> axis image
>> %plot the surface
>> [XX,YY]=meshgrid(0:0.05:2);
>> ZZ=0.25*(XX.^2-YY.^2);
>> subplot(1,2,2)
>> surf(XX,YY,ZZ)
>> shading interp
>> hold on
```

```
>> %add the gradint vector
>> Z=0.25*(X.^2-Y.^2);
>> quiver3(X,Y,Z,U,V,W)
>> axis image
```



مثال دیگر: مطلوبست رسم بردارهای گرادیان رویه $z = xe^{-x^2-y^2}$

```
>>[X,Y] = meshgrid(-2:0.25:2,-1:0.2:1);
>>Z = X.* exp(-X.^2 - Y.^2);
>>[U,V,W] = surfnorm(X,Y,Z);
>>quiver3(X,Y,Z,U,V,W,0.5);
>>hold on
>>surf(X,Y,Z);
>>colormap hsv
>>view(-35,45)
>>axis ([-2 2 -1 1 -.6 .6])
>>hold off
```



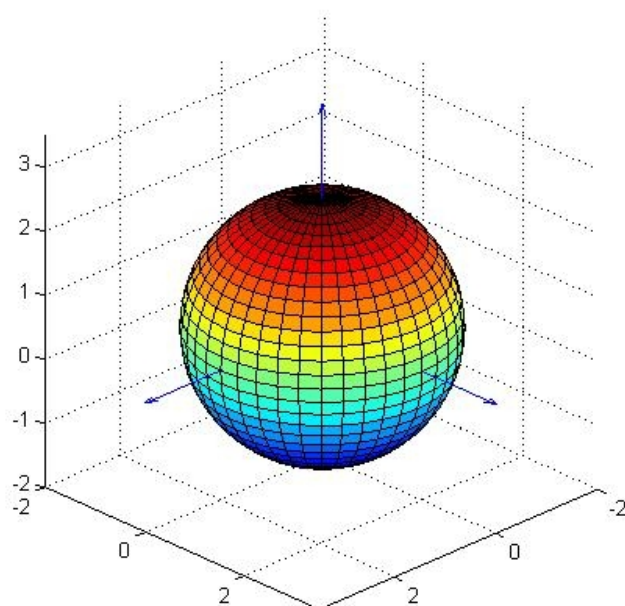
ترسیم پارامتریک سطوح

مثال) می‌خواهیم کره‌ای با مختصات زیر رسم کنیم:

$$x = a \cos(v) \cos(u), \quad y = a \cos(v) \sin(u), \quad z = a \sin(v)$$

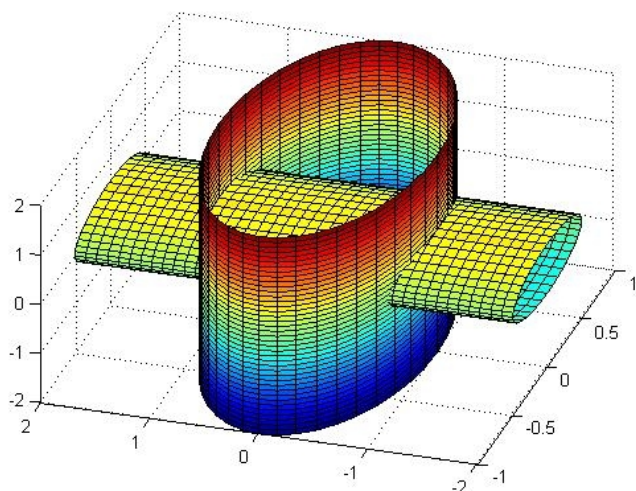
$$0 \leq u \leq 2\pi, \quad -\frac{\pi}{2} \leq v \leq \frac{\pi}{2}$$

```
>> a=2;
>> u=linspace(0,2*pi,41);
>> v=linspace(-pi/2,pi/2,31);
>> [U,V]=meshgrid(u,v);
>> X=a*cos(V).*cos(U);
>> Y=a*cos(V).*sin(U);
>> Z=a*sin(V);
>> surf(X,Y,Z)
>> axis image
```



مثال) نمایش تقاطع دو استوانه

```
>> u=linspace(0,2*pi,41);
>> v=linspace(-2,2,41);
>> [U,V]=meshgrid(u,v);
>> %vertical cylinder with r=1
>> surf(cos(U),sin(U),V)
>> hold on
>> %horizontal cylinder with r=5
>> surf(0.5*cos(U),V,0.5*sin(U))
>> hold off
```



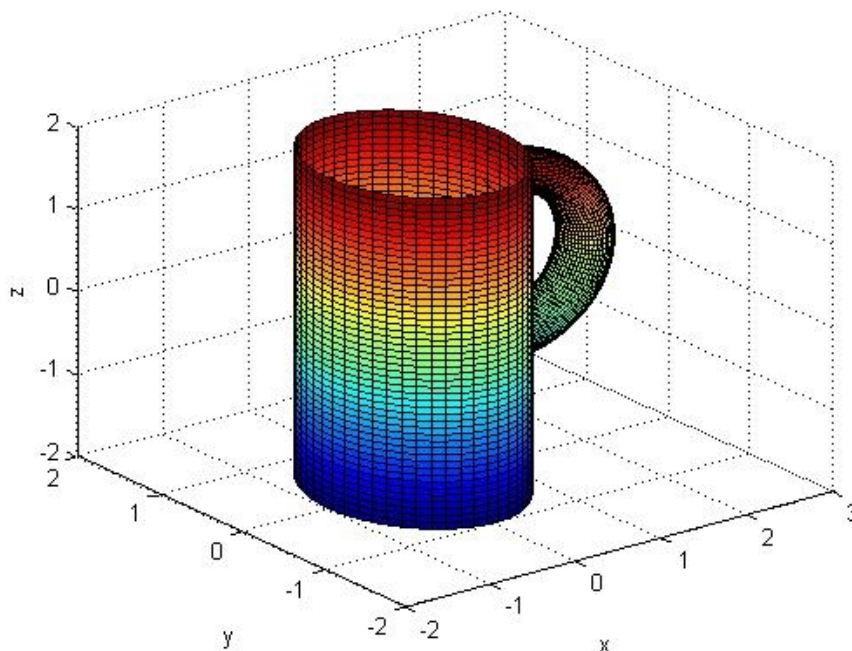
استفاده از فرمان ezsurf برای ترسیم سطوح پارامتری

این فرمان به صورت $\text{ezsurf}(x,y,z,[a \ b \ c \ d])$ می باشد. پارامترها s و t می باشند که $a < s < b$ و $c < t < d$ است.

مثال) ترسیم یک لیوان :

```
>> syms s t
>> %vertical cylinder with r=1
>> x=cos(s);
>> y=sin(s);
>> z=t;
>> ezsurf(x,y,z,[0 2*pi -2 2])
>> hold on
>> %handle r=1 centered in (1,0,0.5)
>> xhandle=1+cos(s)*(1+0.25*cos(t));
>> yhandle=0.25*sin(t);
>> zhandle=0.5+sin(s)*(1+0.25*cos(t));
>> ezsurf(xhandle,yhandle,zhandle,[-pi/2 pi/2 0 2*pi])
>> hold off
>> axis([-2 3 -2 2 -2 2])
```

$x = 1 + \cos(s) (1 + 1/4 \cos(t))$, $y = 1/4 \sin(t)$, $z = 1/2 + \sin(s) (1 + 1/4 \cos(t))$



معادلات دیفرانسیل مرتبه اول

هرچند که MATLAB یک نرم افزار مخصوص مسایل آنالیز عددی می باشد ولی قادر است مسایل مربوط به معادلات دیفرانسیل را به طور نمادین حل نماید. به عنوان مثال می خواهیم معادله دیفرانسیل $y' = xy$ را حل کنیم. تذکر: مشتقات y را باید به صورت اپراتور وارد نمود. تابع مورد استفاده در این مبحث، $\text{dsolve}()$ می باشد که مخصوص حل نمادین معادلات می باشد. مثال مطرح شده را این گونه حل می کنیم:

```
>> syms x y
>> y=dsolve('Dy=y*x','x')
y =
C1*exp(1/2*x^2)
```

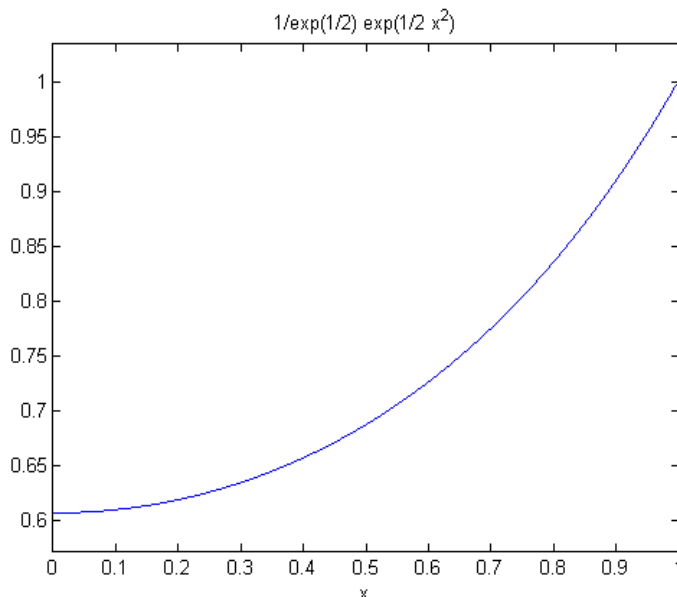
حل معادلات دیفرانسیل با شرایط مرزی

کافی است شرایط مرزی را در تابع dsolve وارد نماییم.

```
>> syms x y
>> y=dsolve('Dy=y*x','y(1)=1','x')
y =
1/exp(1/2)*exp(1/2*x^2)
```

حالا می خواهیم تابع y که در بالا به دست آمد را در بازه $[0, 1]$ رسم کنیم:

```
>> ezplot(y,[0 1])
```



- برای آشنایی بیشتر به حل چند معادله می پردازیم:

```
Ex.) y' = 1 + x + y^2 + xy^2
>> syms xy
>> y=dsolve('Dy=1+x+y^2+x*y^2','x')
y =
```

$$\tan(x+1/2*x^2+C1)$$

Ex.) $y' = \tan(x+y) - 1$

```
>> syms xy
>> y=dsolve('Dy=tan(x+y)-1','x')
y =
-x+asin(exp(x)/C1)
```

Ex.) $y' = \frac{x^2 - y^2}{2xy}$

```
>> syms xy
>> y=dsolve('Dy=(x^2-y^2)/(2*x*y)','x')
```

```
y =
1/3/x*3^(1/2)*(x*(x^3+3*C1))^(1/2)
-1/3/x*3^(1/2)*(x*(x^3+3*C1))^(1/2)
```

Ex.) $y' = \frac{x+2y-5}{3x-y-1}$

```
>> syms xy
>> y=dsolve('Dy=(x+2*y-5)/(3*x-y-1)','x')
y =
-1/2*log((( -1+x)^2+(-y+2)*(-1+x)+(-y+2)^2)/(-1+x)^2)-
5/3*3^(1/2)*atan(1/3*(3+x-2*y)*3^(1/2)/(-1+x))-log(-1+x)-C1 =
0
```

تذکر: در تمامی موارد بالا بعد از حل معادله برای نمایش بهتر y می توان دستور `>>pretty(y)` را اجرا نمود.

Ex.) $xy' - 3y = x^2$

```
>> syms xy
>> y=dsolve('x*(Dy)-3*y=x^2','x')
y =
(-1+C1*x)*x^2
```

Ex.) $y' - y = xy^2$

```
>> syms x y
>> y=dsolve('Dy-y=x*y^2','x')
y =
-1/(-1+x-exp(-x)*C1)
```

Ex.) $y' = \frac{2x}{x^2 \cos(y) + \sin(2y)}$

```
>> syms x y
>> y=dsolve('Dy=(2*x)/(x^2*cos(y)+sin(2*y))','x')
y =
```

$-\text{asin}(\text{lambertw}(-1/2*C1*\exp(-1/2*x^2-1))+1/2*x^2+1)$

که تابع w لامبرت (Lambert's W function) به صورت زیر می باشد :

$W = \text{lambertw}(X)$:

$$we^w = x$$

Ex.) $y' = x^3 + \frac{2}{x}y - \frac{1}{x}y^2$

>> syms x y

>> y=dsolve('Dy=x^3+(2/x)*y-(1/x)*y^2','x')

y =

$$-i*\tan(1/2*i*x^2-C1)*x^2$$

Ex.) $y' \cos(y) + \sin(y) = x + 1$

>> syms x y

>> y=dsolve('(Dy)*cos(y)+sin(y)=x+1','x')

y =

$$\text{asin}((\exp(x)*x-C1)/\exp(x))$$

Ex.) $(4xy + 3y^2 - 1)dx + (x^2 + 2xy)dy = 0$

>> syms x y

>> y=dsolve('4*x*y+3*y^2-1+(x^2+2*x*y)*(Dy)=0','x')

y =

$$\begin{aligned} & -1/6*(3*x^3-(9*x^6+12*x^4-36*C1*x)^(1/2))/x^2 \\ & -1/6*(3*x^3+(9*x^6+12*x^4-36*C1*x)^(1/2))/x^2 \end{aligned}$$

Ex.) $(y + \ln(x))dx - xdy = 0$

>> syms x y

>> y=dsolve('(y+log(x))=x*Dy','x')

y =

$$-\log(x)-1+C1*x$$

Ex.) $y' = -\sqrt{x-a}$

>> syms x y a

>> y=dsolve('Dy=-(x-a)^0.5','x')

y =

$$-2/3*(x-a)^(3/2)+C1$$

Ex.) $xy\rho^2 + (x^2 + xy + y^2)\rho + x^2 + xy = 0$

>> syms x y

>> y=dsolve('x*y*(Dy)^2+(x^2+x*y+y^2)*(Dy)+x^2+x*y=0','x')

y =

$$\begin{aligned} & (1/2*5^(1/2)-1/2)*x \\ & (-x^2+C1)^(1/2) \end{aligned}$$

$$\begin{aligned} & (-1/2*5^{(1/2)}-1/2)*x \\ & -(-x^2+C1)^{(1/2)} \\ & (-x^2+C1)^{(1/2)} \end{aligned}$$

$$Ex.) 16x^2 + 2\rho^2y - \rho^3x = 0$$

```
>> syms x y
>> y=dsolve('16*x^2+2*((Dy)^2)*(y)-((Dy)^3)*x=0','x')
y =
3/2*(-4*x)^(1/3)*x
```

معادلات دیفرانسیل مرتبه دوم

در این حالت نیز همانند معادلات مرتبه اول از تابع dsolve استفاده می کنیم:

$$Ex.) y'' - y' - 6y = 0$$

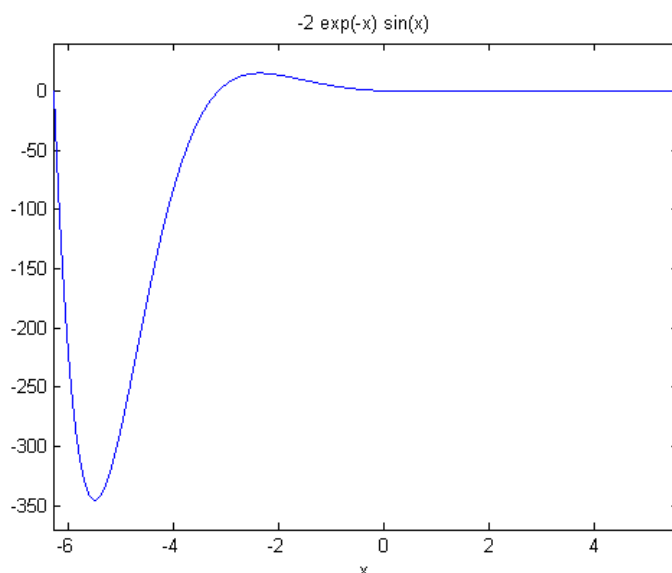
```
>> syms x y
>> y=dsolve('D2y-Dy-6*y=0','x')
y =
C1*exp(3*x)+C2*exp(-2*x)
```

$$Ex.) y'' + 2y' + 2y = 0, \quad y(0)=0, \quad y'(0)=-2$$

```
>> syms x y
>> y=dsolve('D2y+2*Dy+2*y=0','y(0)=0','Dy(0)=-2','x')
y =
-2*exp(-x)*sin(x)
```

حال تابع y را رسم می نمایم:

```
>> ezplot(y)
```



$$Ex.) y'' - 2y' + y = 8 + 2xe^x$$

```
>> syms x y
>> y=dsolve('D2y-2*Dy+y=8+2*x*exp(x)','x')
y =
```

$$8 + 1/3 \exp(x) * (3 * C2 + 3 * x * C1 + x^3)$$

$$Ex.) y'' + 2y' + y = 4e^{-x} \ln(x)$$

```
>> syms x y
```

```
>> y=dsolve('D2y+2*Dy+y=4*(exp(-x))*log(x)', 'x')
```

```
y =
```

$$\exp(-x) * (C2 + x * C1 + 2 * x^2 * \log(x) - 3 * x^2)$$

$$Ex.) x^2 y'' + x y' - 4y = 0$$

```
>> syms x y
```

```
>> y=dsolve('(D2y)*x^2+x*(Dy)-4*y=0', 'x')
```

```
y =
```

$$(C1 + C2 * x^4) / x^2$$

$$Ex.) x^2 y'' - 4xy' + 6y = \frac{1}{x^4}$$

```
>> syms x y
```

```
>> y=dsolve('(D2y)*x^2-4*x*(Dy)+6*y=1/(x^4)', 'x')
```

```
y =
```

$$1/42 * (42 * x^7 * C2 + 42 * x^6 * C1 + 1) / x^4$$

معادلات مرتبه n-ام

این حالت نیز عیناً مانند حالات قبل می باشد :

$$Ex.) (D^4 + 8D^2 + 16)y = -\sin(x)$$

```
>> syms x y
```

```
>> y=dsolve('D4y+8*D2y+16*y=-sin(x)', 'x')
```

```
y =
```

```
-
```

$$1/9 * \sin(x) + C1 * \sin(2 * x) + C2 * \cos(2 * x) + C3 * \sin(2 * x) * x + C4 * \cos(2 * x) * x$$

$$Ex.) xy''' + y'' = x + 1$$

```
>> syms x y
```

```
>> y=dsolve('x*D3y+D2y=x+1', 'x')
```

```
y =
```

$$1/12 * x^3 + C1 * x * \log(x) - C1 * x + 1/2 * x^2 + C2 * x + C3$$

تذکر مهم: در Matlab منظور از $\log(x)$ همان $\ln(x)$ می باشد و $\log(x)$ را در Matlab به صورت

$\log_{10}(x)$ وارد می کنند.

$$Ex.) x^2 y'' - 2xy' + 2y = (\ln(x))^2 - \ln(x^2)$$

```
>> syms x y
```

```
>> y=dsolve('x^2*D2y-2*x*Dy+2*y=(log(x))^2-log(x^2)', 'x')
```

```
y =
```

$$C2*x^2+C1*x-1/2*\log(x^2)+1/4+1/2*\log(x)^2+3/2*\log(x)$$

$$Ex.) (x^2 - 2x)y'' + 4(x - 1)y' + 2y = e^{2x}$$

```
>> syms x y
```

```
>> y=dsolve('(x^2-2*x)*D2y+4*(x-1)*Dy+2*y=exp(2*x)', 'x')
```

```
y =
```

$$1/4*(4*C2*x+4*C1+\exp(2*x))/x/(x-2)$$

$$Ex.) (x + 1)y'' - xy' + y = 0$$

```
>> syms x y
```

```
>> y=dsolve('(x-1)*D2y-x*Dy+y=0', 'x')
```

```
y =
```

$$C1*x+\exp(x)*C2$$

معادله دیفرانسیل ژاندر

معادله دیفرانسیل ژاندر را در حالت کلی بررسی می نمایم :

$$(1 - x^2)y'' - 2xy' + n(n + 1)y = 0$$

```
>> syms x y n
```

```
>> y=dsolve('(1-x^2)*D2y-2*x*Dy+(n)*(n+1)*y=0', 'x')
```

```
y =
```

$$C1*LegendreP(n,x)+C2*LegendreQ(n,x)$$

پس مشاهده می شود که این تابع در Matlab تعریف شده می باشد و به صورت زیر می باشد :

$$P_n^m(x) = (-1)^m (1 - x^2)^{\frac{m}{2}} \frac{d^m}{dx^m} P_n(x)$$

و $P_n(x)$ تابع چندجمله ای ژاندر از مرتبه n می باشد و به صورت زیر است :

$$P_n(x) = \frac{1}{2^n n!} \left[\frac{d^n}{dx^n} (x^2 - 1)^n \right]$$

معادله دیفرانسیل بسل

ابتدا معادله را به صورت کلی وارد می کنیم :

$$x^2 y'' + xy' + (x^2 - v^2)y = 0$$

```
>> syms x y v
```

```
>> y=dsolve('x^2*D2y+x*Dy+(x^2-v^2)*y=0', 'x')
```

```
y =
```

$$C1*besselj(v,x)+C2*bessely(v,x)$$

مشاهده می گردد که تابع بسل در Matlab تعریف شده می باشد :

تابع بسل نوع اول

این نوع بسل مربوط به معادله زیر می باشد :

$$x^2 y'' + xy' + (x^2 - v^2)y = 0 \quad \& \quad v \in \mathbb{R}$$

در این حالت تابع بسل به صورت زیر تعریف می گردد :

$$J_v(x) = \left(\frac{x}{2}\right)^v \sum_{k=0}^{\infty} \frac{\left(-\frac{x^2}{4}\right)^k}{k! \Gamma(v+k+1)}$$

اگر بخواهیم تابع بسل را در یک نقطه حساب کنیم دستور آن به صورت زیر می باشد :

```
J = besselj(v,x)
```

مثال : می خواهیم مقدار $J_5(9)$ را حساب کنیم :

```
>> J=besselj(5,9)
J =
-0.0550
```

تابع بسل نوع دوم

این نوع بسل مربوط به معادله زیر می باشد :

$$x^2 y'' + xy' - (x^2 - v^2)y = 0 \quad \& \quad v \in \mathbb{R}$$

در این حالت تابع بسل به صورت زیر تعریف می گردد :

$$K_v(x) = \left(\frac{\pi}{2}\right) \frac{I_{-v}(x) - I_v(x)}{\sin(v\pi)}$$

به طوریکه :

$$I_v(x) = \left(\frac{x}{2}\right)^v \sum_{k=0}^{\infty} \frac{\left(\frac{x^2}{4}\right)^k}{k! \Gamma(v+k+1)}$$

اگر بخواهیم تابع بسل نوع دوم را در یک نقطه حساب کنیم دستور آن به صورت زیر می باشد :

```
>>K= besselk(v,x)
```

تابع بسل نوع سوم

این نوع بسل مربوط به معادله زیر می باشد و به تابع Hankel نیز موسوم می باشد :

$$x^2 y'' + xy' + (x^2 - v^2)y = 0 \quad \& \quad v \geq 0$$

در این حالت تابع بسل به صورت زیر تعریف می گردد :

$$Y_v(x) = \frac{J_v(x) \cos(v\pi) - J_{-v}(x)}{\sin(v\pi)}$$

به طوریکه $Y_v(x)$ تابع $bessely$ می باشد .

رابطه این دو نوع بسل به شکل زیر می باشد :

$$H_v^{(1)}(x) = J_v(x) + iY_v(x)$$

$$H_v^{(2)}(x) = J_v(x) - iY_v(x)$$

اگر بخواهیم این نوع تابع بسل را در یک نقطه حساب کنیم دستور آن به صورت زیر می باشد :

```
>>H = besselh(v,x)
```

می باشد. $k=1$ که

دستگاه معادلات دیفرانسیل خطی

در این حالت باز هم از دستور dsolve استفاده می شود، با این تفاوت که تمامی معادلات را وارد دستور می نماییم.

$$Ex.) \begin{cases} y_1' = 2y_1 - 5y_2 \\ y_2' = 5y_1 - 6y_2 \end{cases}$$

```
>> syms y1 y2
```

```
>> [y1 y2]=dsolve('Dy1=2*y1-5*y2','Dy2=5*y1-6*y2')
```

y1 =

```
exp(-2*t)*(sin(3*t)*C1+cos(3*t)*C2)
```

y2 =

```
1/5*exp(-2*t)*(4*sin(3*t)*C1-  
3*cos(3*t)*C1+4*cos(3*t)*C2+3*sin(3*t)*C2)
```

$$26) \begin{cases} \dot{x} - 2x - 3y = 2e^{2t} \\ -x + \dot{y} - 4y = 3e^{2t} \end{cases}$$

```
>> syms x y
```

```
>> [x y]=dsolve('Dx-2*x-3*y=2*exp(2*t)','-x+Dy-  
4*y=3*exp(2*t)')
```

x =

```
-3*exp(t)*C2+exp(5*t)*C1-5/3*exp(2*t)
```

y =

```
exp(t)*C2+exp(5*t)*C1-2/3*exp(2*t)
```

$$Ex.) \begin{cases} \ddot{x} + \dot{x} + x + \ddot{y} + y = e^t \\ \ddot{x} + \dot{x} + \ddot{y} = e^{-t} \end{cases}$$

```
>> syms x y
```

```
>> [x y]=dsolve('D2x+Dx+x+D2y+y=exp(t)','D2x+Dx+D2y=exp(-t)')
```

x =

```
-2*exp(-t)-exp(t)+C1
```

```
y =
exp(-t)+2*exp(t)-C1
```

تبدیل لاپلاس

در این حالت تابع مورد نظر را بر حسب t وارد می کنیم و آن را در ورودی دستور laplace قرار می دهیم.
خروجی دستور بر حسب s می باشد.

Ex.) $F = 4 + t^3 - e^{-3t} + 4 \cos(2t) - 6 \sinh(7t)$

```
>> syms t
>> F=4+t^3-exp(-3*t)+4*cos(2*t)-6*sinh(7*t);
>> f=laplace(F)
f =
4/s+6/s^4-1/(s+3)+4*s/(s^2+4)-42/(s^2-49) >> syms t
```

Ex.) $F = t^2 e^{-3t} \cos(at)$

```
>> syms t a
>> F=t^2*exp(-3*t)*cos(a*t);
>> f=laplace(F)
f =
2*(s+3)*(s^2+6*s+9-3*a^2)/(s^2+6*s+9+a^2)^3
```

Ex.) $F = \int_0^t e^{-3u} \cos(4u) du$

```
>> syms t u
>> I=int(exp(-3*u)*cos(4*u),u,0,t);
>> f=laplace(I)
f =
(s+3)/s/(s^2+6*s+25)
```

Ex.) $F = t \int_0^t u^2 e^{-3u} \sin(4u) du$

```
>> syms t u
>> F=t*int(u^2*exp(-3*u)*sin(4*u),u,0,t);
>> f=laplace(F)
f =8*(275+264*s+326*s^2+144*s^3+15*s^4)/s^2/(s^2+6*s+25)^4
```

تذکر: برای نمایش بهتر خروجی می توان دستور `>>pretty(f)` را اجرا نمود.

Ex.) $F = t e^{3t} \int_0^t e^{4u} \cos(4u) du$

```
>> syms t u
>> F=t*exp(3*t)*int(exp(4*u)*cos(6*u),u,0,t);
>> f=laplace(F)
```

$$f = \frac{2(s^3 - 19s^2 + 119s - 317)}{(s-3)^2(s^2 - 14s + 85)^2}$$

تبدیل لاپلاس معکوس

در این حالت تابع مورد نظر را بر حسب s وارد می کنیم و آن را در ورودی دستور `ilaplace` قرار می دهیم.

خروجی دستور بر حسب t می باشد:

$$Ex.) f = \frac{3}{s^7} - \frac{2}{s-3} + \frac{4}{s^2+3} - \frac{s}{s^2+2}$$

```
>> syms s
>> f=3/(s^7)-2/(s-3)+4/(s^2+3)-s/(s^2+2);
>> F=ilaplace(f)
F =
1/240*t^6-2*exp(3*t)+4/3*3^(1/2)*sin(3^(1/2)*t)-
cos(2^(1/2)*t)
```

$$Ex.) f = \frac{s+4}{(s-3)(s+2)(s-6)}$$

```
>> syms s
>> f=(s+4)/((s-3)*(s+2)*(s-6));
>> F=ilaplace(f)
F =
1/20*exp(-2*t)+5/12*exp(6*t)-7/15*exp(3*t)
```

$$Ex.) f = \frac{4s+3}{2s^2+3s+4}$$

```
>> syms s
>> f=(4*s+3)/(2*s^2+3*s+4);
>> F=ilaplace(f)
F =
2*exp(-3/4*t)*cos(1/4*23^(1/2)*t)
```

$$Ex.) f = \tan^{-1} \frac{1}{s+6}$$

```
>> syms s
>> f=atan(1/(s+6));
>> F=ilaplace(f)
F =
exp(-6*t)/t*sin(t)
```

$$Ex.) f = \frac{1}{s+4} \tan^{-1} \frac{1}{s+4}$$

```
>> syms s
>> f=(1/(s+4))*atan(1/(s+4));
>> F=ilaplace(f)
F =
```

$\exp(-4*t)*\sinint(t)$

ملاحظه می شود که منظور از $\sinint(t)$ همان سینوس انتگرال می باشد.

$$Ex.) F = \frac{e^{-4s}}{\sqrt{2s^2+9}} \tan^{-1} \frac{1}{s}$$

```
>> syms s
>> F=(exp(-4*s)*atan(1/s))/(2*s^2+9)^0.5;
>> f=ilaplace(F)
f =
 1/2*heaviside(t-
4)*2^(1/2)*int(1/_U1*sin(_U1)*besselj(0,3/2*2^(1/2)*(t-4-
_U1)),_U1 = 0 .. t-4)
```

تابع گاما

تابع گاما به طور پیش فرض به صورت $\gamma(x)$ در Matlab تعریف شده می باشد.

```
Ex.) Γ(5)
>> gamma(10)
ans =
 362880
```

تبدیل فوریه

مثال) تبدیل فوریه توابع زیر را حساب کنید.

$$Ex.) f(x) = e^{-x^2}$$

```
>> syms x
>> f=exp(-x^2);
>> F=fourier(f)
```

F =

$\exp(-1/4*w^2)*\pi^{(1/2)}$

$$Ex.) f(x) = e^{-x|x|}$$

```
>> syms x
>> f=x*exp(-abs(x));
>> F=fourier(f)
```

F =

$-4*i/(1+w^2)^2$

```
>> pretty(F)
```

$$\frac{-4 i w}{(1 + w^2)^2}$$

تبدیل فوریه معکوس

مثال) تبدیل فوریه معکوس تابع زیر را حساب کنید. دقت کنید که ω حقیقی است.

Ex.) $f(\omega) = e^{-|\omega|}$

```
>> syms w real
>> F=exp(-abs(w));
>> f = ifourier(F)
```

f =

$1/(1+x^2)/\pi$

وارد کردن ماتریس

برای وارد کردن درایه‌های ماتریس، مانند مثال زیر عمل می‌کنیم.

$$Ex.) A = \begin{bmatrix} 1 & 2 & -3 \\ 4 & 5 & \pi \\ -0.6 & 9 & \sqrt{2} \end{bmatrix}$$

```
>> A=[1 2 -3;4 5 pi;-0.6 9 sqrt(2)]
```

A =

```
    1.0000    2.0000   -3.0000
    4.0000    5.0000    3.1416
   -0.6000    9.0000    1.4142
```

|

می‌توان به جای به کار بردن space از , استفاده کرد.

```
>> A=[1,2,-3;4,5,pi;-0.6,9,sqrt(2)]
```

A =

```
    1.0000    2.0000   -3.0000
    4.0000    5.0000    3.1416
   -0.6000    9.0000    1.4142
```

```
>> B=[1 3 5;-1 2 4;0 1 -3]
```

B =

```
    1    3    5
   -1    2    4
    0    1   -3
```

ترانپاده ماتریس

```
>> At=A'
```

At =

```
    1.0000    4.0000   -0.6000
    2.0000    5.0000    9.0000
   -3.0000    3.1416    1.4142
```

دترمینان ماتریس

```
>> detA=det(A)
```

detA =

```
-153.2869
```

اثر (trace) ماتریس

برای محاسبه اثر (trace) ماتریس، یعنی جمع عناصر روی قطر اصلی از دستور trace استفاده می‌کنیم.

```
>> trace(A)
```

```
ans =
```

```
7.414213562373095
```

معکوس ماتریس

```
>> A_invers=inv(A)
```

```
A_invers =
```

```
0.1383    0.1946   -0.1388
0.0492    0.0025    0.0988
-0.2544    0.0665    0.0196
```

عملیات روی ماتریس‌ها

```
>> C=A+B
```

```
C =
```

```
2.0000    5.0000    2.0000
3.0000    7.0000    7.1416
-0.6000   10.0000   -1.5858
```

```
>> D=A*B
```

```
D =
```

```
-1.0000    4.0000   22.0000
-1.0000   25.1416   30.5752
-9.6000   17.6142   28.7574
```

```
>> E=B^4
```

```
E =
```

```
-40    40   -104
  0    -8   -168
 -8   -32   176
```

```
>> F=A+3.75*B
```

```
F =
```

```
    4.7500    13.2500    15.7500
    0.2500    12.5000    18.1416
   -0.6000    12.7500   -9.8358
```

تولید ماتریس واحد، قطری، یک و صفر

```
>> G=eye(4)
```

```
G =
```

```
    1    0    0    0
    0    1    0    0
    0    0    1    0
    0    0    0    1
```

```
>> G1=eye(4,3)
```

```
G1 =
```

```
    1    0    0
    0    1    0
    0    0    1
    0    0    0
```

```
>> G1=eye(4,5)
```

```
G1 =
```

```
    1    0    0    0    0
    0    1    0    0    0
    0    0    1    0    0
    0    0    0    1    0
```

```
>> H=[1 2 -3 -4];  
>> K=diag(H)
```

```
K =
```

```
    1    0    0    0  
    0    2    0    0  
    0    0   -3    0  
    0    0    0   -4
```

```
>> diag(K)
```

```
ans =
```

```
    1  
    2  
   -3  
   -4
```

```
>> ans'
```

```
ans =
```

```
    1    2   -3   -4
```

```
>> L=ones(4)
```

```
L =
```

```

1     1     1     1
1     1     1     1
1     1     1     1
1     1     1     1
```

```
>> L1=ones(4,6)
```

```
L1 =
```

```

1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
1     1     1     1     1     1
```

```
>> M=zeros(4,6)
```

```
M =
```

```

0     0     0     0     0     0
0     0     0     0     0     0
0     0     0     0     0     0
0     0     0     0     0     0
```

تولید ماتریس با درایه‌های تصادفی توزیع یکنواخت

```
>> N=rand(4,6)
```

```
N =
```

```

0.8147    0.6324    0.9575    0.9572    0.4218    0.6557
0.9058    0.0975    0.9649    0.4854    0.9157    0.0357
0.1270    0.2785    0.1576    0.8003    0.7922    0.8491
0.9134    0.5469    0.9706    0.1419    0.9595    0.9340
```

تولید ماتریس با درایه‌های تصادفی توزیع نرمال

```
>> M=randn(4,6)
```

```
M =
```

```

-0.4326   -1.1465    0.3273   -0.5883    1.0668    0.2944
-1.6656    1.1909    0.1746    2.1832    0.0593   -1.3362
 0.1253    1.1892   -0.1867   -0.1364   -0.0956    0.7143
 0.2877   -0.0376    0.7258    0.1139   -0.8323    1.6236
```

اندیس ماتریس

```
>> a=[1 2 3;4 5 6;7 8 9]
```

```
a =
```

1	2	3
4	5	6
7	8	9

```
>> x=a(3,2)
```

```
x =
```

```
8
```

```
>> a(2,3)=-9
```

```
a =
```

1	2	3
4	5	-9
7	8	9

```
>> a(4,3)=10
```

```
a =
```

1	2	3
4	5	-9
7	8	9
0	0	10

تولید ماتریس با علامت : (colon)

مثال) برداری تولید کنید که شروع آن، ۱۰ باشد و با گام ۰/۳ تا ۱۴ پیش برود.

```
>> a=10:0.3:14
```

```
a =
```

```
Columns 1 through 8
```

```
10.0000    10.3000    10.6000    10.9000    11.2000    11.5000    11.8000    12.1000
```

```
Columns 9 through 14
```

```
12.4000    12.7000    13.0000    13.3000    13.6000    13.9000
```

```
>> a'
```

```
ans =
```

```
10.0000
```

```
10.3000
```

```
10.6000
```

```
10.9000
```

```
11.2000
```

```
11.5000
```

```
11.8000
```

```
12.1000
```

```
12.4000
```

```
12.7000
```

```
13.0000
```

```
13.3000
```

```
13.6000
```

```
13.9000
```

عملیات روی سطر و ستون ماتریس

```
>> a=[1 2 3;-4 -5 -6;7 8 9]
```

```
a =
```

```
     1     2     3
    -4    -5    -6
     7     8     9
```

```
>> b=a(:,1)
```

```
b =
```

```
     1
    -4
     7
```

```
>> b=a(:,2)
```

```
b =
```

```
     2
    -5
     8
```

```
>> b'
```

```
ans =
```

```
     2    -5     8
```

```
>> c=a(3,:)
```

```
c =
```

```
     7     8     9
```

```
>> d=a(1:2,2:3)
```

```
d =
```

```
     2     3
    -5    -6
```

```
>> a
```

```
a =
```

1	2	3
-4	-5	-6
7	8	9

```
>> a(2,:)=[4.5 5.5 6.5]
```

```
a =
```

1.0000	2.0000	3.0000
4.5000	5.5000	6.5000
7.0000	8.0000	9.0000

مشاهده سائز ماتریس

```
>> size(a)
```

```
ans =
```

```

      3      3
>> e=[1 2 3 4 5 6];
>> length(e)
```

```
ans =
```

```
6
```

تابع sum

این تابع، ستون‌های ماتریس را جمع می‌کند.

```
>> a=[1 2 3;4 5 6;7 8 9]
```

```
a =
```

1	2	3
4	5	6
7	8	9

```
>> s=sum(a)
```

```
s =
```

12	15	18
----	----	----

تابع repmat

برای تولید ماتریس با تکرار کردن ماتریس مورد نظر به کار می‌رود.

```
>> a=[1 2 3 4]
```

```
a =
```

```
1     2     3     4
```

```
>> b=repmat(a,2,3)
```

```
b =
```

```
1     2     3     4     1     2     3     4     1     2     3     4
1     2     3     4     1     2     3     4     1     2     3     4
```

```
>> c=[1 2 3;4 5 6]
```

```
c =
```

```
1     2     3
4     5     6
```

```
>> d=repmat(c,2,3)
```

```
d =
```

```
1     2     3     1     2     3     1     2     3
4     5     6     4     5     6     4     5     6
1     2     3     1     2     3     1     2     3
4     5     6     4     5     6     4     5     6
```

تولید بردار با رابطه خطی

دستور linspace(a,b,n) یک بردار با شروع a و خاتمه b با تعداد n عضو تولید می‌کند.

```
>> a=linspace(2,20,5)
```

```
a =
```

```
2.0000    6.5000   11.0000   15.5000   20.0000
```

```
>> b=linspace(-2,3,8)
```

```
b =
```

```
-2.0000   -1.2857   -0.5714    0.1429    0.8571    1.5714    2.2857    3.0000
```

تولید بردار با رابطه لگاریتمی

دستور $\text{logspace}(a, b, n)$ یک بردار با شروع 10^a و خاتمه 10^b با تعداد n عضو تولید می کند.

```
>> c=logspace(1,5,5)
```

c =

10	100	1000	10000	100000
----	-----	------	-------	--------

```
>> d=[linspace(1,5,5),linspace(-2,3,3);linspace(-21,3,5),linspace(2.5,3.1,3);linspace(8,10,5),linspace(4,12,3)]
```

d =

1.0000	2.0000	3.0000	4.0000	5.0000	-2.0000	0.5000	3.0000
-21.0000	-15.0000	-9.0000	-3.0000	3.0000	2.5000	2.8000	3.1000
8.0000	8.5000	9.0000	9.5000	10.0000	4.0000	8.0000	12.0000

تذکر: هرگاه دستور طولانی شد، می توان با قراردادن ... ادامه آن را در سطر بعدی نوشت.

```
>> d=[linspace(1,5,5),linspace(-2,3,3) ...
```

```
;linspace(-21,3,5),linspace(2.5,3.1,3);linspace(8,10,5),linspace(4,12,3)]
```

d =

1.0000	2.0000	3.0000	4.0000	5.0000	-2.0000	0.5000	3.0000
-21.0000	-15.0000	-9.0000	-3.0000	3.0000	2.5000	2.8000	3.1000
8.0000	8.5000	9.0000	9.5000	10.0000	4.0000	8.0000	12.0000

حذف درایه های ماتریس

```
>> a=[1 2 3;4 5 6;7
```

a =

1	2	3
4	5	6
7	8	9

```
>> a(:,2)=[]
```

a =

1	3
4	6
7	9

```
>> b=[1 2 3 5 6 -2 -3];
```

```
>> b(4)=[]
```

b =

1	2	3	6	-2	-3
---	---	---	---	----	----

عملیات بر روی آرایه‌ها

ماتریس زیر را در نظر بگیرید:

$$Ex.) A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

می‌خواهیم هر کدام از عناصر را به توان ۲ برسانیم. اگر دستور A^2 را اجرا کنیم، ماتریس A در خودش ضرب می‌شود.

پس باید عملگر دیگری را تعریف کنیم.

در عملیات بر روی آرایه‌ها به جای $*$ از $.*$ و به جای $^$ از $.^$ استفاده می‌کنیم.

```
>> A=[1 2 3;4 5 6;7 8 9]
```

```
A =
```

```

1     2     3
4     5     6
7     8     9
```

```
>> B=[0.1 0.2 0.3;0.4 0.7 0.9;0.2 -0.7 -20]
```

```
B =
```

```

0.1000    0.2000    0.3000
0.4000    0.7000    0.9000
0.2000   -0.7000  -20.0000
```

```
>> A.^2
```

```
ans =
```

```

1     4     9
16    25    36
49    64    81
```

```
>> A.*B
```

```
ans =
```

```

0.1000    0.4000    0.9000
1.6000    3.5000    5.4000
1.4000   -5.6000 -180.0000
```

```
>> A./B
```

```
ans =
```

```

10.0000    10.0000    10.0000
10.0000     7.1429     6.6667
35.0000   -11.4286    -0.4500
```

```
>> A.^B

ans =

    1.0000    1.1487    1.3904
    1.7411    3.0852    5.0158
    1.4758    0.2333    0.0000
```

بدین وسیله می‌توانیم ماتریس‌ها را در ورودی توابع مورد نظر قرار دهیم.

```
>> a=[1 2 3;4 5 6;7 8 9];
>> f=inline('x.^2+2*cos(x)', 'x')

f =

    Inline function:
    f(x) = x.^2+2*cos(x)

>> f(a)

ans =

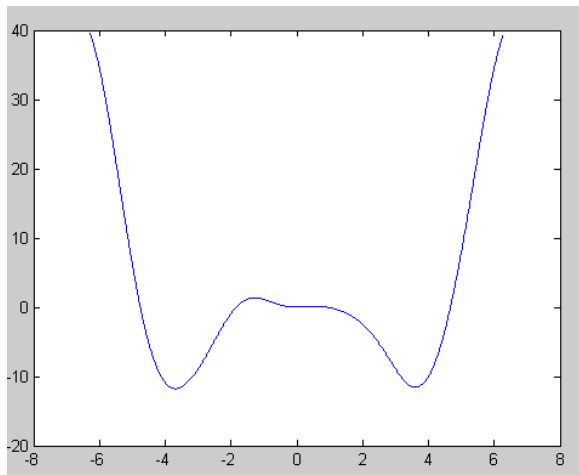
    2.0806    3.1677    7.0200
   14.6927   25.5673   37.9203
   50.5078   63.7090   79.1777
```

رسم نمودار توابع یک متغیره با دستور plot

دستور plot نقاط x_i و y_i را رسم می‌کند.

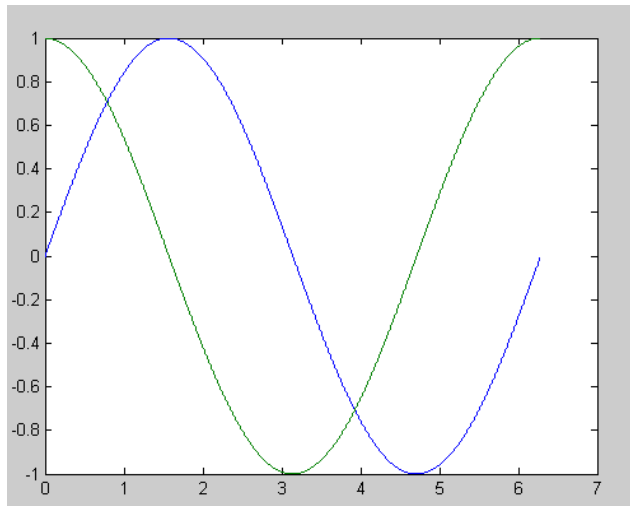
مثال) نمودار تابع زیر را در فاصله $-2\pi \leq x < 2\pi$ رسم کنید.

```
y = f(x) = x^2 cos x + sin^3 x
>> x=-2*pi:0.01:2*pi;
>> y=(x.^2).*cos(x)-(sin(x)).^3;
>> plot(x,y)
```



مثال) رسم نمودار $\sin(x)$ و $\cos(x)$ در کنار هم.

```
>> x=0:0.001:2*pi;
>> y1=sin(x);
>> y2=cos(x);
>> plot(x,y1,x,y2)
```

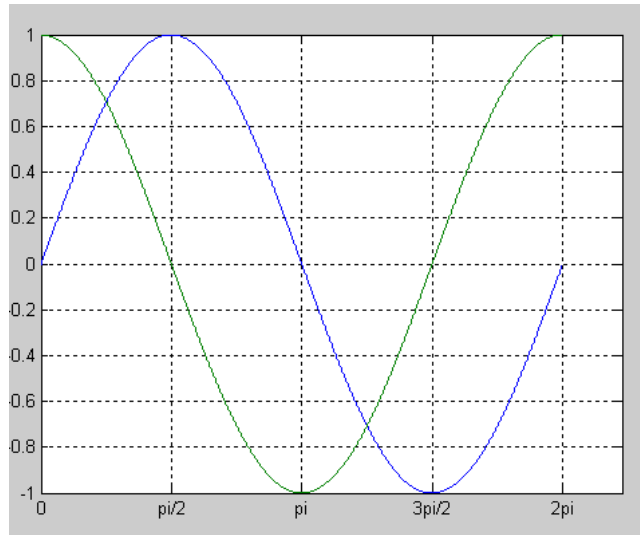


برای حل این روش می‌توان فرامین زیر را نیز به کار برد.

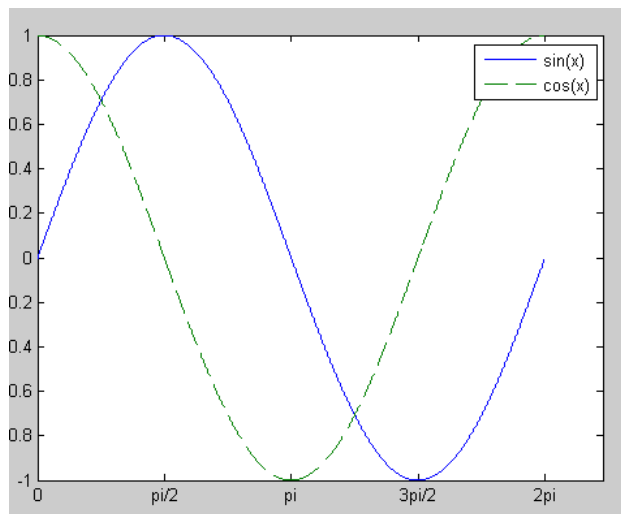
```
>> x=0:0.001:2*pi;
>> y1=sin(x);
>> y2=cos(x);
>> plot(x,y1)
>> hold on
>> plot(x,y2)
```

به دستورات زیر دقت کنید.

```
>> x=0:0.001:2*pi;
>> y1=sin(x);
>> y2=cos(x);
>> plot(x,y1,x,y2)
>> set(gca,'XTick',0:pi/2:2*pi)
>> set(gca,'XTickLabel',{'0','pi/2','pi','3pi/2','2pi'})
>> grid
```



```
>> x=0:0.001:2*pi;
>> y1=sin(x);
>> y2=cos(x);
>> plot(x,y1,'-',x,y2,'--')
>> legend('sin(x)','cos(x)')
>> set(gca,'XTick',0:pi/2:2*pi)
>> set(gca,'XTickLabel',{'0','pi/2','pi','3pi/2','2pi'})
```

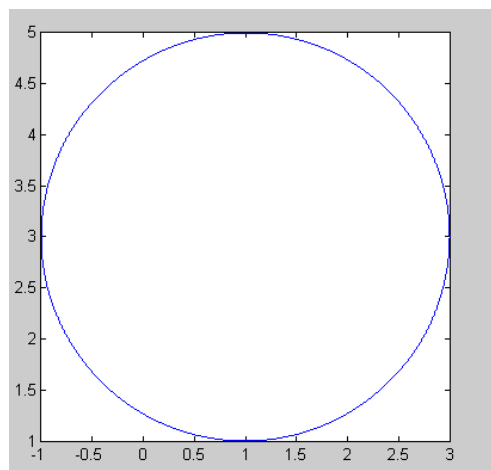


رسم پارامتریک منحنی در صفحه با plot

مثال) مطلوبیت رسم منحنی زیر.

$$Ex.) (x-1)^2 + (y-3)^2 = 2$$

```
>> t=0:0.001:2*pi;
>> x=1+2*cos(t);
>> y=3+2*sin(t);
>> plot(x,y)
>> axis square
```



رسم پارامتریک منحنی در فضا با plot3

مثال) مطلوبست رسم مارپیچ دایره‌ای با معادله زیر.

$$x = \cos 2t, y = \sin 2t, z = t, 0 \leq t \leq 8\pi$$

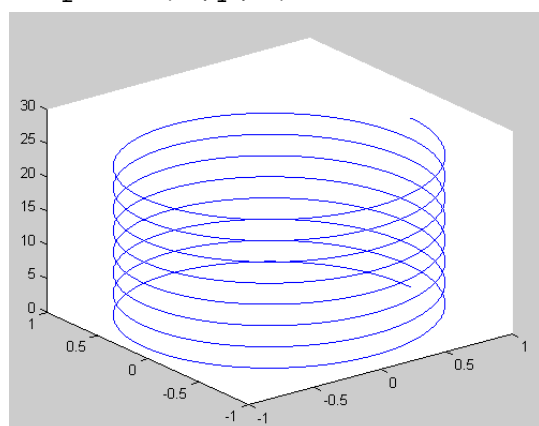
```
>> t=0:0.001:8*pi;
```

```
>> x=cos(2*t);
```

```
>> y=sin(2*t);
```

```
>> z=t;
```

```
>> plot3(x,y,z)
```



دستور reshape

```
>> reshape(a,6,2)
```

```
ans =
```

```

1     3
5     7
9    11
2     4
6     8
10    12
```

```
>> b=-2:2:24
```

```
b =
```

```
Columns 1 through 13
```

```
-2      0      2      4      6      8     10     12     14     16     18     20     22
```

```
Column 14
```

```
24
```

```
>> c=reshape(b,7,2)
```

```
c =
```

```
-2      12
 0      14
 2      16
 4      18
 6      20
 8      22
10      24
```

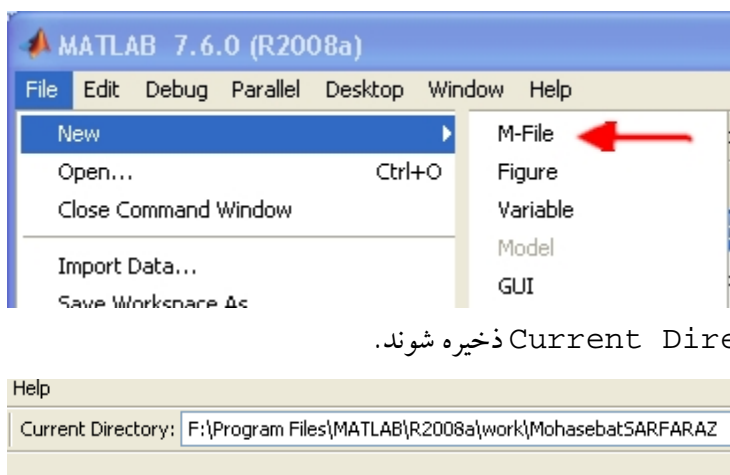
M-File ها

به سه دلیل نیاز به ایجاد M-File داریم:

- ۱- ایجاد و ذخیره توابع پیچیده تر
- ۲- نوشتن و ضبط توالی زیاد فرامین
- ۳- استفاده از ساختارهای کنترل و تکرار

دو نوع M-File داریم: M-File function و M-File script

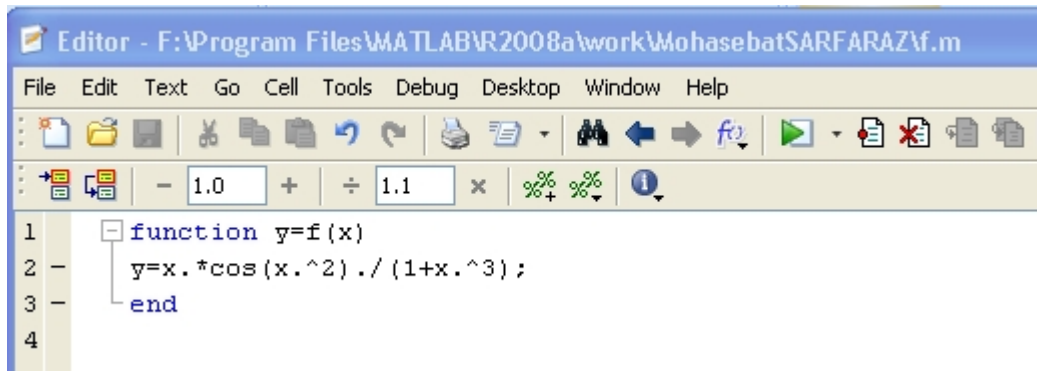
برای نوشتن و ایجاد M-File به صورت زیر عمل می کنیم.



توجه شود که M-File ها باید در Current Directory ذخیره شوند.

function M-File

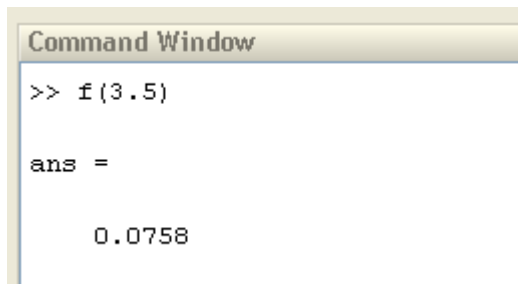
فرض کنیم که نیاز به محاسبه تابع $y=f(x)=x\cos(x^2)/(1+x^3)$ داریم. M-File زیر را ایجاد و ذخیره می‌کنیم و برای به دست آوردن مقدار تابع در $x=3.5$ کافی است در خط فرمان دستور $f(3.5)$ را اجرا کنیم.



```

1 function y=f(x)
2     y=x.*cos(x.^2)./(1+x.^3);
3 end
4

```



```

>> f(3.5)

ans =

    0.0758

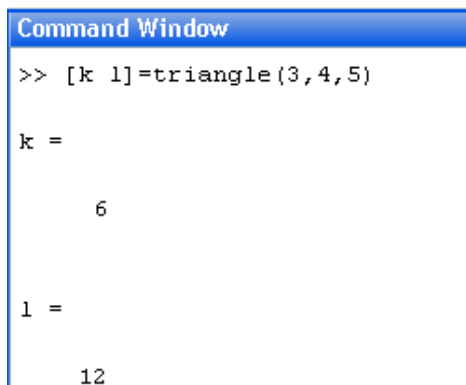
```

ممکن است یک تابع چندین ورودی و خروجی داشته باشد. به عنوان مثال می‌خواهیم محیط و مساحت یک مثلث با سه ضلع معلوم را حساب کنیم.

```

function [area prim]=triangle(a,b,c)
z=(a+b+c)/2;
area=sqrt(z*(z-a)*(z-b)*(z-c));
prim=2*z;
end

```



```

>> [k l]=triangle(3,4,5)

k =

    6

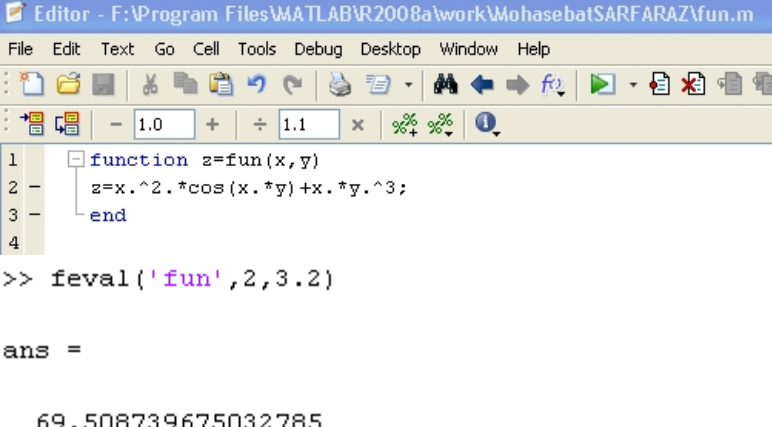
l =

   12

```

دستور feval

کاربرد این دستور در M-File ها محاسبه مقدار تابع در ازای ورودی های آن است.
(مثال)



```

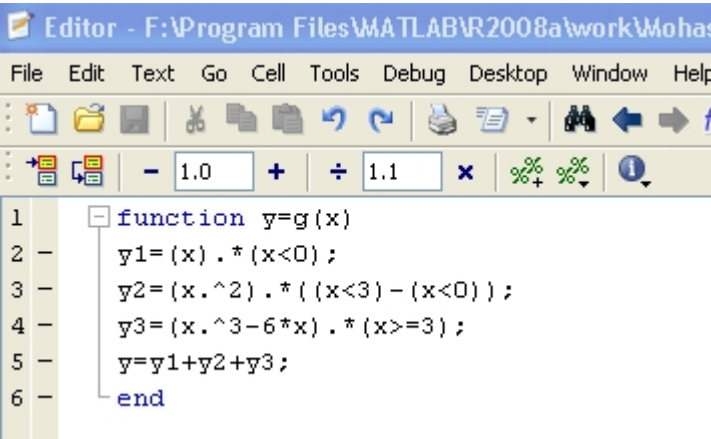
Editor - F:\Program Files\MATLAB\R2008a\work\MohasebatSARFARAZ\fun.m
File Edit Text Go Cell Tools Debug Desktop Window Help
+ - 1.0 + ÷ 1.1 × % + % - !
1 function z=fun(x,y)
2 z=x.^2.*cos(x.*y)+x.*y.^3;
3 end
4
>> feval('fun',2,3.2)

ans =

69.508739675032785
  
```

استفاده از عبارات منطقی برای محاسبه توابع چند ضابطه ای

$$\text{Ex.) } g(x) = \begin{cases} x & x < 0 \\ x^2 & 0 \leq x < 3 \\ x^3 - 6x & x \geq 3 \end{cases}$$



```

Editor - F:\Program Files\MATLAB\R2008a\work\Mohas
File Edit Text Go Cell Tools Debug Desktop Window Help
+ - 1.0 + ÷ 1.1 × % + % - !
1 function y=g(x)
2 y1=(x).*(x<0);
3 y2=(x.^2).*((x<3)-(x<0));
4 y3=(x.^3-6*x).*(x>=3);
5 y=y1+y2+y3;
6 end
  
```

```
>> g(0)
```

```
ans =
```

```
0
```

```
>> g(3)
```

```
ans =
```

```
9
```

```
>> g(2)
```

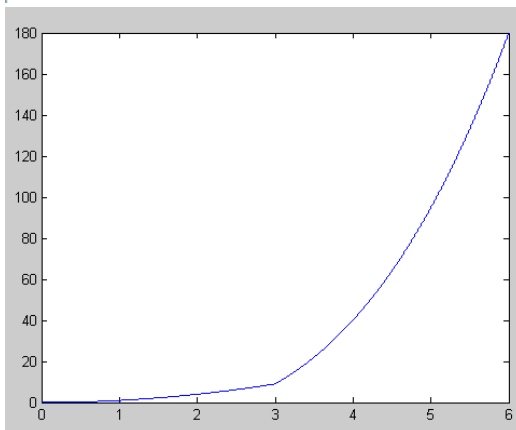
```
ans =
```

```
4
```

برای رسم نمودار $y=g(x)$ در فاصله 0 تا 6 به صورت زیر عمل می‌کنیم.

Command Window

```
>> x=0:0.0001:6;
>> y=g(x);
>> plot(x,y)
>>
```



script M-File

این نوع M-File برای جمع‌آوری مجموعه‌ای از فرامین که یک برنامه را می‌سازند، استفاده می‌شود و مانند تابع، ورودی نمی‌گیرد و فقط نقش اجرایی دارد.

مثال) می‌خواهیم نمودار $y=f(x)=x^n e^{-nx}$ را برای $n=1, 2, \dots, 10$ در فاصله 0 تا 20 رسم کنیم.

```

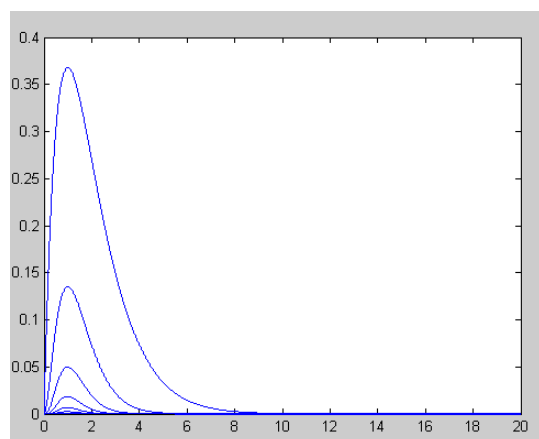
Editor - F:\Program Files\MATLAB\R2008a\work\MohasebatSARFARAZ\graph.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons]
- 1.0 + ÷ 1.1 × %>% %>% ⓘ
1 - clear
2 - clc
3 - x=0:0.001:20;
4 - for n=1:10
5 -     y=x.^n.*exp(-n*x);
6 -     plot(x,y)
7 -     hold on
8 - end

```

M-File را با نام graph.m ذخیره می کنیم.

Command Window

```
>> graph
```



ریاضیات عددی

حل معادلات غیر خطی به روش نصف کردن فاصله‌ها

مثال صفحه ۳۴:

$$x^3 + x^2 - 3x - 3 = 0 \quad x \in (1,2)$$

M-File شماره ۱:

```
[c,err,yc]=bisect(f,a,b,delta)
```

c: ریشه تابع، err: خطایی که برنامه را قطع کرده، yc: مقدار تابع در ازای ریشه یعنی $f(c)$ ، f: تابع مورد نظر

که به صورت inline وارد می‌شود، a: نقطه ابتدایی بازه، b: نقطه انتهایی بازه، delta: خطای قابل اغماض

Command Window

```
>> f=inline('x^3+x^2-3*x-3','x');
>> [c,err,yc]=bisect(f,1,2,0.00001)
```

```
k =
```

```
17
```

```
c =
```

```
1.732051849365234
```

```
err =
```

```
7.629394531250000e-006
```

```
yc =
```

```
9.859673310685935e-006
```

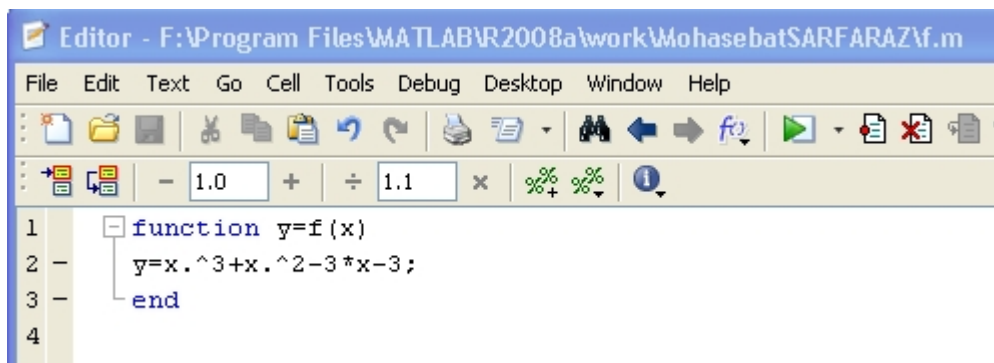
مشاهده می‌شود که ۱۷ بار عمل تکرار صورت گرفته است.

M-File شماره ۲:

```
r = bisect2(fun,[a,b],xtol,ftol,verbose)
```

r: ریشه تابع، f: تابع مورد نظر که به صورت M-File وارد می‌شود، a: نقطه ابتدایی بازه، b: نقطه انتهایی

بازه، xtol: خطای قابل اغماض، اگر verbose برابر ۱ باشد، تکرار را نشان می‌دهد.



```

Editor - F:\Program Files\MATLAB\R2008a\work\MohasebatSARFARAZf.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 function y=f(x)
2 y=x.^3+x.^2-3*x-3;
3 end
4

```

```
>> r = bisect2('f',[1,2],0.00001,0.00001,1)
```

Bisection iterations for f.m

k	xm	fm
1	1.5000e+000	-1.8750e+000
2	1.7500e+000	1.7188e-001
3	1.6250e+000	-9.4336e-001
4	1.6875e+000	-4.0942e-001
5	1.7188e+000	-1.2479e-001
6	1.7344e+000	2.2030e-002
7	1.7266e+000	-5.1755e-002
8	1.7305e+000	-1.4957e-002
9	1.7324e+000	3.5127e-003
10	1.7314e+000	-5.7282e-003
11	1.7319e+000	-1.1092e-003
12	1.7322e+000	1.2013e-003
13	1.7321e+000	4.5962e-005
14	1.7320e+000	-5.3166e-004
15	1.7320e+000	-2.4286e-004
16	1.7320e+000	-9.8448e-005
17	1.7320e+000	-2.6243e-005

```
r =
```

```
1.732048034667969
```

```
>> r = bisect2('f',[1,2],0.00001,0.00001,0)
```

```
r =
```

```
1.732048034667969
```

حل معادلات غیر خطی به روش درونیابی خطی

```
[p1,err,k,y]=secant(f,p0,p1,delta,epsilon,max1)
```

p1: ریشه تابع، f: تابع مورد نظر که به صورت M-File وارد می شود، p0: نقطه ابتدایی بازه، p1: نقطه انتهایی

بازه، xtol: خطای قابل اغماض، k: تعداد تکرار، y برابر f(p1)، max1: حداکثر تکرار

(مثال صفحه ۳۴)

$$x^3 + x^2 - 3x - 3 = 0 \quad x \in (1,2)$$

```
>> f=inline('x^3+x^2-3*x-3','x');
>> [p1,err,k,y]=secant(f,1,2,0.00001,0.00001,100)
```

p1 =

1.732050697785584

err =

5.432700288432457e-005

k =

5

y =

-1.039000173008731e-006

تذکر مهم: اگر در امتحان ریشه ها را در تکرارهای مختلف خواستند، باید تغییر کوچکی در برنامه بدهیم، بدین صورت که بعد از خط ۲۱ عبارت p1 را اضافه و سپس M-File را ذخیره می کنیم و برنامه را دوباره run می کنیم.

```
19 - relerr=2*err/(abs(p2)+delta);
20 - p0=p1;
21 - p1=p2;
22 - p1 ←
23 - y=feval(f,p1);
24 - if (err<delta) || (relerr<delta)
25 - end
```

حل معادلات غیر خطی به روش درونیابی خطی اصلاح شده

```
[k x3]=modifsecant(f,a,b,epsilon,max1)
```

x3: ریشه تابع، f: تابع مورد نظر که به صورت M-File وارد می شود، a: نقطه ابتدایی بازه، b: نقطه انتهایی بازه،

epsilon: خطای قابل اغماض، k: تعداد تکرار، max1: حداکثر تکرار

(مثال صفحه ۳۴)

$$x^3 + x^2 - 3x - 3 = 0 \quad x \in (1,2)$$

```
>> f=inline('x^3+x^2-3*x-3','x');
>> [k x3]=modifsecant(f,1,2,0.00001,100)
```

k =

7

x3 =

1.732050807477299

حل معادلات غیر خطی به روش تکرار تابعی

```
[k,p,err,P] = fixpt(g,p0,tol,max1)
```

x3: ریشه تابع، g: تابع مورد نظر که به صورت M-File وارد می شود، p0: نقطه شروع، tol: خطای قابل

اغماض، k: تعداد تکرار، max1: حداکثر تکرار، P: بردار شامل مقادیر p در هر مرحله، err: تفاوت pها در دو تکرار

متوالی

(مثال صفحه ۴۳)

$$g(x) = \sqrt{2x+3}, x_1 = 4$$

```
>> g=inline('sqrt(2*x+3)','x');
>> [k,p,err,P] = fixpt(g,4,0.00001,500)
```

k =

12

p =

3.000005225941382

```

err =

    1.045189641990518e-005

P =

    4.000000000000000
    3.316624790355400
    3.103747667048789
    3.034385495301739
    3.011440019426500
    3.003810919291193
    3.001270037597814
    3.000423315999865
    3.000141102014992
    3.000047033636303
    3.000015677837802
    3.000005225941382

```

حل معادلات غیر خطی به روش نیوتن

مثال صفحه ۵۰

$$f(x) = 3x + \sin x - e^x = 0 \quad x_1 = 0.0$$

M-File شماره ۱:

```
[p0,err,k,y]=newton(f,df,p0,delta,epsilon,max1)
```

p0: ریشه تابع، err: خطایی که برنامه را قطع کرده، y: مقدار تابع در ازای ریشه یعنی $f(p0)$ ، f: تابع مورد نظر که به صورت inline وارد می‌شود، df: مشتق تابع مورد نظر که به صورت inline وارد می‌شود، p0 ورودی: نقطه شروع، delta: خطای قابل اغماض برای p0، k: تعداد تکرار، epsilon: خطای قابل اغماض برای y، max1 حداکثر تکرار

```

>> f=inline('3*x+sin(x)-exp(x)','x');
>> df=inline('3+cos(x)-exp(x)','x');
>> [p0,err,k,y]=newton(f,df,0,0.00001,0.00001,500)

```

```

p0 =

    0.360421680476020

err =

    2.509668983860514e-004

```

k =

3

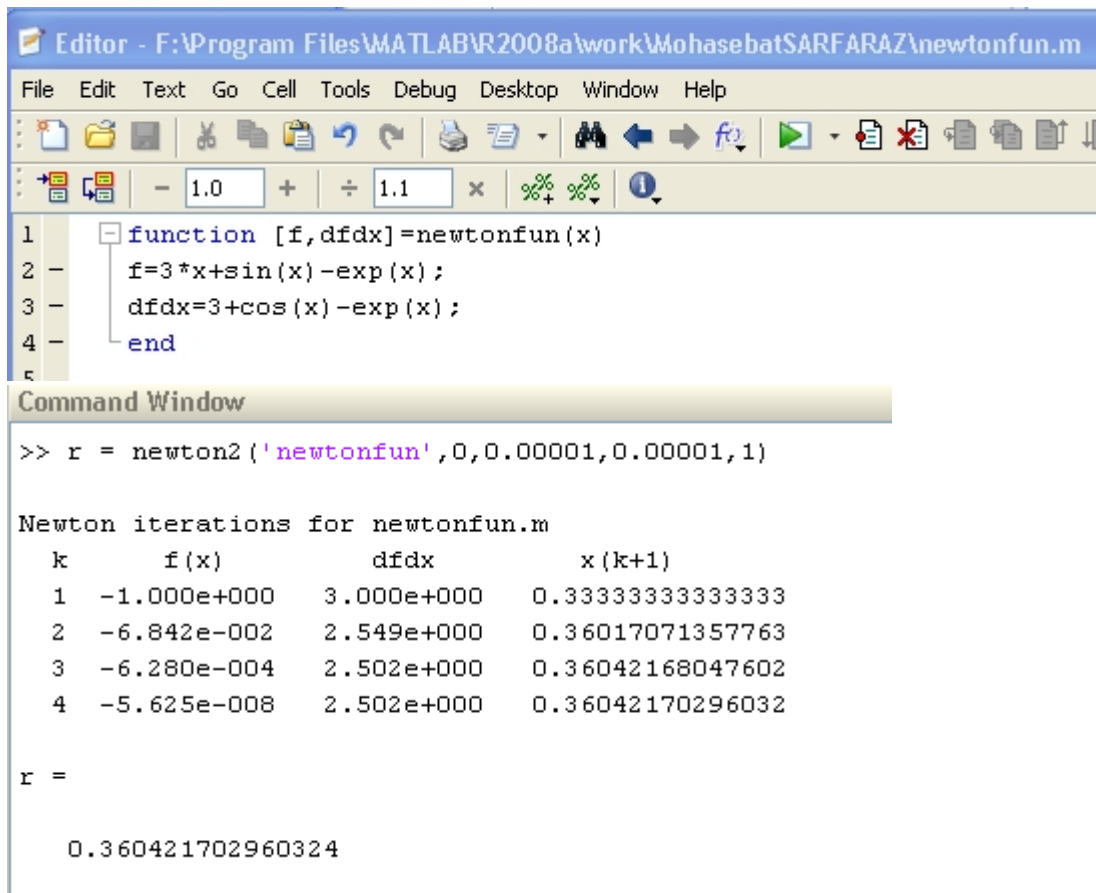
y =

-5.625155319322062e-008

:M-File شماره ۲:

r = newton2('newtonfun',x0,xtol,ftol,verbose)

r : ریشه تابع، newtonfun: مجموعه توابع مورد نظر که به صورت M-File وارد می شود، x0: نقطه شروع، xtol: خطای قابل اغماض، اگر verbose برابر ۱ باشد، تکرار را نشان می دهد.



The screenshot shows the MATLAB Editor window with the file 'newtonfun.m' open. The function code is as follows:

```

1 function [f,dfdx]=newtonfun(x)
2     f=3*x+sin(x)-exp(x);
3     dfdx=3+cos(x)-exp(x);
4 end

```

Below the editor, the Command Window shows the execution of the function:

```

>> r = newton2('newtonfun',0,0.00001,0.00001,1)

Newton iterations for newtonfun.m
    k      f(x)      dfdx      x(k+1)
    1   -1.000e+000   3.000e+000   0.333333333333333
    2   -6.842e-002   2.549e+000   0.36017071357763
    3   -6.280e-004   2.502e+000   0.36042168047602
    4   -5.625e-008   2.502e+000   0.36042170296032

r =

    0.360421702960324

```

حل معادلات غیر خطی به روش مولر

```
[p,y,err,k]=muller(f,p0,p1,p2,delta,epsilon,max1)
```

p: ریشه تابع، err: خطایی که برنامه را قطع کرده، y: مقدار تابع در ازای ریشه یعنی $f(p_0)$ ، f: تابع مورد نظر که به صورت inline وارد می‌شود، df: مشتق تابع مورد نظر که به صورت inline وارد می‌شود، p0 و p1 و p2: مقادیر اولیه، delta: خطای قابل اغماض، max1 حداکثر تکرار، k: تعداد تکرار

مثال صفحه ۶۰

$$f(x) = \sin x - \frac{x}{2} = 0, x_0 = 2.0, x_1 = 2.2, x_2 = 1.8$$

Command Window

```
>> f=inline('sin(x)-x/2','x');
>> [p,y,err,k]=muller(f,2,2.2,1.8,0.00001,0.00001,500)

p =

    1.895494424968581

y =

   -1.293520067724430e-007

err =

    2.423176810005002e-004

k =

     2
```

حل دستگاه معادلات خطی به روش حذفی گاوس - جردن

M-File شماره ۱ (بدون پایداری):

`x = GEshow(A,b,ptol)`x: بردار جواب، A و b ماتریس‌هایی که در رابطه $Ax=b$ هستند، ptol: خطای قابل اغماض

مثال صفحه ۱۰۱)

$$\begin{bmatrix} 1 & 3 & 4 \\ 2 & 7 & 18 \\ 7 & 1 & 3 \end{bmatrix} x = \begin{bmatrix} 2 \\ 1 \\ 5 \end{bmatrix}$$

`>> A=[1 3 4;2 7 18;7 1 3];``>> b=[2;1;5];``>> x = GEshow(A,b,50*eps)`

Begin forward elimination with Augmented system:

1	3	4	2
2	7	18	1
7	1	3	5

After elimination in column 1 with pivot = 1.000000

1	3	4	2
0	1	10	-3
0	-20	-25	-9

After elimination in column 2 with pivot = 1.000000

1	3	4	2
0	1	10	-3
0	0	175	-69

x =

```

0.748571428571428
0.942857142857143
-0.394285714285714

```

`>> format rat``>> x`

x =

```

131/175
33/35
-69/175

```

M-File شماره ۲ (با پایداری، سطرهای بزرگ را در اول می آورد):

`x = GEPivShow(A,b,ptol)`

x: بردار جواب، A و b ماتریس هایی که در رابطه $Ax=b$ هستند، ptol: خطای قابل اغماض

مثال صفحه ۱۰۳:

$$\begin{bmatrix} 0.6 & 3.8 & 7 \\ 2.6 & 3.1 & 5 \\ 3.7 & 5.8 & 2.9 \end{bmatrix} x = \begin{bmatrix} 5.7 \\ 2.6 \\ 3.4 \end{bmatrix}$$

```
>> A=[.6 3.8 7;2.6 3.1 5;3.7 5.8 2.9];
```

```
>> b=[5.7;2.6;3.4];
```

```
>> x = GEPivShow(A,b,50*eps)
```

Begin forward elimination with Augmented system:

0.6000000000000000	3.8000000000000000	7.0000000000000000	5.7000000000000000
2.6000000000000000	3.1000000000000000	5.0000000000000000	2.6000000000000000
3.7000000000000000	5.8000000000000000	2.9000000000000000	3.4000000000000000

Swap rows 1 and 3; new pivot = 3.7

After elimination in column 1 with pivot = 3.700000

3.7000000000000000	5.8000000000000000	2.9000000000000000	3.4000000000000000
0	-0.975675675675676	2.962162162162162	0.210810810810811
0	2.859459459459460	6.529729729729730	5.148648648648649

Swap rows 2 and 3; new pivot = 2.85946

After elimination in column 2 with pivot = 2.859459

3.7000000000000000	5.8000000000000000	2.9000000000000000	3.4000000000000000
0	2.859459459459460	6.529729729729730	5.148648648648649
0	0	5.190170132325141	1.967580340264650

x =

```
-0.843695367132867
0.934877622377622
0.379097465034965
```

روش سوم)

در این روش از M-File ای استفاده نمی شود.

مثال صفحه ۱۰۳)

$$\begin{bmatrix} 0.6 & 3.8 & 7 \\ 2.6 & 3.1 & 5 \\ 3.7 & 5.8 & 2.9 \end{bmatrix} x = \begin{bmatrix} 5.7 \\ 2.6 \\ 3.4 \end{bmatrix}$$

```
>> A=[.6 3.8 7;2.6 3.1 5;3.7 5.8 2.9];
>> b=[5.7;2.6;3.4];
>> x=A\b
```

x =

```
-0.843695367132867
0.934877622377623
0.379097465034965
```

حل دستگاه معادلات خطی به روش تجزیه ماتریس به L و U

روش اول) در این روش از دستوری که خود برنامه دارد، استفاده می‌کنیم.

مثال صفحه ۱۰۵)

$$A = \begin{bmatrix} 3 & -1 & 2 \\ 1 & 2 & 3 \\ 2 & -2 & -1 \end{bmatrix}$$

```
>> A=[3 -1 2;1 2 3;2 -2 -1];
>> [L U]=lu(A)
```

L =

```
1.0000    0    0
0.3333    1.0000    0
0.6667   -0.5714    1.0000
```

U =

```
3.0000   -1.0000    2.0000
0    2.3333    2.3333
0    0   -1.0000
```

```
>> format rat
```

```
>> L
```

L =

```
1    0    0
1/3    1    0
2/3   -4/7    1
```

```
>> U
```

U =

```
3    -1    2
0    7/3    7/3
0    0   -1
```

مثال صفحه ۱۰۶

$$\begin{bmatrix} 3.7 & 5.8 & 2.9 \\ 2.6 & 3.1 & 5 \\ 0.6 & 3.8 & 7 \end{bmatrix} x = \begin{bmatrix} 3.4 \\ 2.6 \\ 5.7 \end{bmatrix}$$

```
>> A=[3.7 5.8 2.9;2.6 3.1 5;.6 3.8 7];
>> b=[3.4;2.6;5.7];
>> [L U]=lu(A);
>> z=L\b
z =
```

```
3.400000000000000
5.148648648648649
1.967580340264650
```

```
x =
```

```
-0.843695367132867
0.934877622377623
0.379097465034965
```

روش دوم

```
X = lufact(A,B)
```

مثال صفحه ۱۰۶

$$\begin{bmatrix} 3.7 & 5.8 & 2.9 \\ 2.6 & 3.1 & 5 \\ 0.6 & 3.8 & 7 \end{bmatrix} x = \begin{bmatrix} 3.4 \\ 2.6 \\ 5.7 \end{bmatrix}$$

```
>> A=[3.7 5.8 2.9;2.6 3.1 5;.6 3.8 7];
>> b=[3.4;2.6;5.7];
>> X = lufact(A,b)
```

```
X =
```

```
-0.843695367132867
0.934877622377622
0.379097465034965
```

روش سوم)

 $[L,U] = \text{luNopiv}(A, \text{ptol})$

(مثال صفحه ۱۰۵)

$$A = \begin{bmatrix} 3 & -1 & 2 \\ 1 & 2 & 3 \\ 2 & -2 & -1 \end{bmatrix}$$

```
>> A=[3 -1 2;1 2 3;2 -2 -1];
>> [L,U] = luNopiv(A,50*eps)
```

L =

1	0	0
1/3	1	0
2/3	-4/7	1

U =

3	-1	2
0	7/3	7/3
0	0	-1

حل دستگاه معادلات خطی به روش تجزیه چولسکی

باید توجه داشته باشیم که ماتریس مورد نظر، متقارن باشد.

روش اول) استفاده از M-File

C = Cholesky(A)

(مثال)

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{bmatrix}$$

```
>> A=[1 1 1 1;1 2 3 4;1 3 6 10;1 4 10 20];
>> C = Cholesky(A)
```

C =

1	1	1	1
0	1	2	3
0	0	1	3
0	0	0	1

روش دوم) استفاده از دستور موجود در Matlab

>> C=chol(A)

C =

1	1	1	1
0	1	2	3
0	0	1	3
0	0	0	1

مثال صفحه ۱۱۱ به روش چولسکی)

$$\begin{bmatrix} 7 & -4 & 0 \\ -4 & 12 & -6 \\ 0 & -6 & 14 \end{bmatrix} x = \begin{bmatrix} 12 \\ 0 \\ 0 \end{bmatrix}$$

>> A=[7 -4 0;-4 12 -6;0 -6 14];

>> b=[12;0;0];

>> c=chol(A);

>> x=c\ (c'\b)

x =

```
2.262857142857143
0.960000000000000
0.411428571428571
```

حل دستگاه معادلات خطی به روش ژاکوبی

باید توجه کرد که ماتریس A حتما قطری مسلط باشد.

$$|a_{ii}| \geq \sum_{j=1}^n |a_{ij}|, i \neq j$$

M-File شماره ۱:

X=jacobi(A,B,P,delta,max1)

P ماتریس حدس اولیه است.

(مثال صفحه ۱۱۱)

$$\begin{bmatrix} 7 & -4 & 0 \\ -4 & 12 & -6 \\ 0 & -6 & 14 \end{bmatrix} x = \begin{bmatrix} 12 \\ 0 \\ 0 \end{bmatrix}, x^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

>> A=[7 -4 0;-4 12 -6;0 -6 14];

>> b=[12;0;0];

>> X=jacobi(A,b,[1;1;1],0.00001,500)

X =

```
2.262848599656562
0.960001910979077
0.411422164028136
```

M-File شماره ۲:

```

X=jacobi2(A,B,P,delta,max1)

>> A=[7 -4 0;-4 12 -6;0 -6 14];
>> b=[12;0;0];
>> X=jacobi2(A,b,[1;1;1],0.00001,500)

Marhaleye [1]
    2.285714285714286
    0.833333333333333
    0.428571428571429

Marhaleye [2]
    2.190476190476191
    0.976190476190476
    0.357142857142857

Marhaleye [3]
    2.272108843537415
    0.908730158730159
    0.418367346938775

Marhaleye [4]
    2.233560090702948
    0.966553287981859
    0.389455782312925

.
.
.

Marhaleye [22]
    2.262848599656562
    0.960001910979077
    0.411422164028136

X =

    2.262848599656562
    0.960001910979077
    0.411422164028136

>> X=jacobi2(A,b,[1;1;1],eps,500)

```

اگر خطای قابل اغماض را برابر eps قرار دهیم

```
Marhaleye [77]
2.262857142857143
0.9600000000000000
0.411428571428571
```

X =

```
2.262857142857143
0.9600000000000000
0.411428571428571
```

حل دستگاه معادلات خطی به روش گاوس - سیدل

باید توجه کرد که ماتریس A حتما قطری مسلط باشد.

$$|a_{ii}| \geq \sum_{j=1}^n |a_{ij}|, i \neq j$$

M-File شماره ۱:

```
X=gseid(A,B,P,delta,max1)
```

P ماتریس حدس اولیه است.

(مثال صفحه ۱۱۲)

```
[ 7  -4  0 ]
[-4  12 -6] x = [12] , x^(0) = [1]
[ 0  -6 14]      [ 0]      [ 1]

>> A=[7 -4 0;-4 12 -6;0 -6 14];
>> b=[12;0;0];
>> X=gseid(A,b,[1;1;1],0.00001,500)
```

X =

```
2.262865384766905
0.960005838019415
0.411431073436892
```

M-File شماره ۲:

```
X=gseid2(A,B,P,delta,max1)

>> A=[7 -4 0;-4 12 -6;0 -6 14];
>> b=[12;0;0];
>> X=gseid2(A,b,[1;1;1],0.00001,500)
```

Marhaleye [1]

```
2.285714285714286
1.261904761904762
0.540816326530612
```

Marhaleye [2]

```
2.435374149659864
1.082199546485261
0.463799805636540
```

```
.
.
.
```

Marhaleye [12]

```
2.262877505222436
0.960014423342083
0.411434752860893
```

Marhaleye [13]

```
2.262865384766905
0.960005838019415
0.411431073436892
```

X =

```
2.262865384766905
0.960005838019415
0.411431073436892
```

حل دستگاه معادلات غیر خطی به روش نیوتن

```
x = newtonSys(Jfun,x0,xtol,ftol,maxit,verbose,varargin)
```

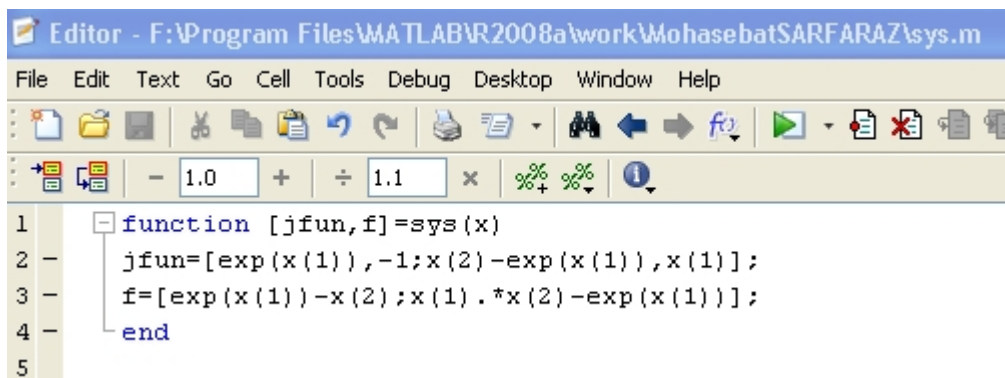
(مثال صفحه ۱۱۸)

$$\begin{cases} e^x - y = 0 \\ xy - e^x = 0 \end{cases}$$

نقاط اولیه $x_0 = 0.95$ و $y_0 = 2.7$ می‌باشند. مشتق‌های جزئی را حساب می‌کنیم.

$$\begin{cases} f_x = e^x \\ f_y = -1 \\ g_x = y - e^x \\ g_y = x \end{cases}$$

ابتدا یک M-File به شکل زیر ایجاد و با نام sys.m ذخیره می‌کنیم. دقت شود که به جای x از x(1) و به جای y از x(2) استفاده شده است.



```
1 function [jfun,f]=sys(x)
2     jfun=[exp(x(1)), -1; x(2)-exp(x(1)), x(1)];
3     f=[exp(x(1))-x(2); x(1).*x(2)-exp(x(1))];
4     end
5
```

سپس در خط فرمان، دستور زیر را اجرا می‌کنیم.

```
>> x = newtonSys('sys', [0.95;2.7], 0.00001, 0.00001, 500, 1)
```

Newton iterations

k	norm(f)	norm(dx)
1	1.162e-001	5.270e-002
2	4.181e-003	2.548e-003
3	8.627e-007	8.971e-007

x =

```
1.00000000000000086
2.718281828459177
```

(مثال)

$$\begin{cases} e^x - xy \sin x + zx = 8.791 \\ \sinh x - xyz + \cos(xz) = -7.533 \\ 2xy + z - 4 \tanh z = 4.151 \end{cases}$$

نقاط اولیه $x_0=1$ و $y_0=2$ و $z_0=3$ می‌باشند. مشتق‌های جزئی را حساب می‌کنیم.

$$\begin{cases} f_x = e^x + y \sin x + xy \cos x + z \\ f_y = x \sin x \\ f_z = x \\ g_x = \cosh x - yz - (\sin xz) z \\ g_y = -zx \\ g_z = -xy - (\sin xz) x \\ h_x = 2y \\ h_y = 2x \\ h_z = -3 + 4 \tanh^2 z \end{cases}$$

ابتدا یک M-File به شکل زیر ایجاد و با نام sys2.m ذخیره می‌کنیم. دقت شود که به جای x از $x(1)$ و به جای y از $x(2)$ و به جای z از $x(3)$ استفاده شده است.

```

Editor - F:\Program Files\MATLAB\R2008a\work\MohasebatSARFARAZ\sys2.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 function [jfun,f]=sys2(x)
2 jfun=[exp(x(1))+x(2)*sin(x(1))+x(1)*x(2)*cos(x(1))+x(3),x(1)*sin(x(1)),x(1);...
3 cosh(x(1))-x(2)*x(3)-sin(x(3)*x(1))*x(3),-x(3)*x(1),-x(1)*x(2)-sin(x(3)*x(1))*x(1);...
4 2*x(2),2*x(1),-3+4*tanh(x(3))^2];
5 f=[exp(x(1))+x(1)*x(2)*sin(x(1))+x(3)*x(1)-8.791;...
6 sinh(x(1))-x(1)*x(2)*x(3)+cos(x(1)*x(3))+7.533;...
7 2*x(1)*x(2)+x(3)-4*tanh(x(3))-4.151];
8 end
9

```

سپس در خط فرمان، دستور زیر را اجرا می‌کنیم.

```
>> x = newtonSys('sys2',[1;2;3],0.0000001,0.0000001,500,1)
```

```
Newton iterations
```

k	norm(f)	norm(dx)
1	2.483e+000	3.363e-001
2	9.963e-002	8.072e-002
3	3.742e-003	9.670e-003
4	6.156e-005	1.336e-004
5	1.141e-008	2.484e-008

```
x =
```

```
1.100104523608455
2.200333077232529
3.298915531163949
```

محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با دستور eig

(مثال صفحه ۱۳۶)

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$$

```
>> A=[1 2;2 4];
```

```
>> [V,D]=eig(A)
```

```
V =
```

```
-0.894427190999916    0.447213595499958
 0.447213595499958    0.894427190999916
```

```
D =
```

```
0    0
0    5
```

اعداد روی قطر اصلی D یعنی 0 و 5 مقادیر ویژه هستند.

ستون‌های V بردارهای ویژه هستند.

مثال صفحه ۱۴۱

$$A = \begin{bmatrix} 2 & 3 & 2 \\ 10 & 3 & 4 \\ 3 & 6 & 1 \end{bmatrix}$$

```
>> A=[2 3 2;10 3 4;3 6 1];
>> [V,D]=eig(A)
```

V =

```
0.371390676354104    0.182574185835055   -0.000000000000000
0.742781352708207    0.365148371670111   -0.554700196225229
0.557086014531156   -0.912870929175277    0.832050294337844
```

D =

```
10.999999999999993    0    0
0    -2.000000000000002    0
0    0    -2.999999999999999
```

محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش لوریه - فادیو

```
[C,s,roots,A_invese]=LevFad(A)
```

C: بردار ضرایب معادله مشخصه، s: معادله مشخصه، roots: مقادیر ویژه

مثال صفحه ۱۴۱

$$A = \begin{bmatrix} 2 & 3 & 2 \\ 10 & 3 & 4 \\ 3 & 6 & 1 \end{bmatrix}$$

```
>> A=[2 3 2;10 3 4;3 6 1];
>> [C,s,roots,A_invese]=LevFad(A)
```

Marhaleye [1]

C(1)= 6.0000

A(1)=

```
2    3    2
10   3    4
3    6    1
```

Marhaleye [2]

C(2)= 49.0000

A(2)=

```
28    9    6
2    45   12
51   -3   25
```

```

Marhaleye [3]
C(3)=      66.0000
A(3)=
    66     0     0
     0    66     0
     0     0    66

C =

     6    49    66

s =

-66-49*x-6*x^2+x^3 = 0
roots =

-2
-3
11
A_invese =

-0.318181818181818    0.136363636363636    0.090909090909091
 0.030303030303030   -0.060606060606061    0.181818181818182
 0.772727272727273   -0.045454545454545   -0.363636363636364

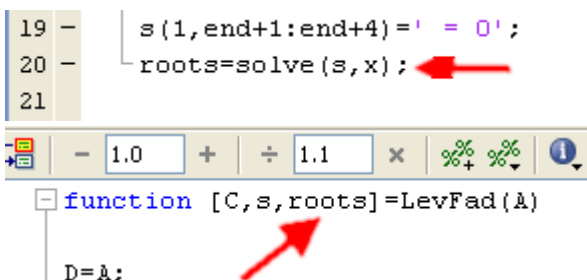
```

تذکر: چون این برنامه از دستور تحلیلی solve برای حل استفاده می کند، ممکن است قادر به حل برخی از معادلات نباشد و در پایان error بدهد. در این صورت باید خط آخر برنامه و root در خروجی تابع، پاک شوند.

```

19 - s(1,end+1:end+4)=' = 0';
20 - roots=solve(s,x);
21

```



```

function [C,s,roots]=LevFad(A)
    D=A;

```

محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش توانی

```
[lambda,v] = powerit(A,s,nit,x0,verbose)
```

lambda: بزرگترین مقدار ویژه، v: بردار ویژه، s: پارامتر shift که در روش توانی، 0 است، nit: حداکثر تکرار، x0: حدس اولیه

مثال صفحه ۱۴۲

$$A = \begin{bmatrix} 2 & 3 & 2 \\ 10 & 3 & 4 \\ 3 & 6 & 1 \end{bmatrix}, z^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

```
>> A=[2 3 2;10 3 4;3 6 1];
>> [lambda,v] = powerit(A,0,10,[1;1;1],1)
```

```
k    norm(u,inf)
1      17.000000
2      9.470588
3     11.583851
4     10.831635
5     11.049800
6     10.985906
7     11.003953
8     10.998903
9     11.000303
10    10.999917
```

```
lambda =
```

```
10.999916852432536
```

```
v =
```

```
0.5000000822104932
1.0000000000000000
0.750003642445965
```

محاسبه مقادیر ویژه و بردارهای ویژه ماتریس با روش تکراری معکوس

```
[D,V]=powerInv(A,z,rho,maxl)
```

D: مقدار ویژه مورد نظر، V: بردار ویژه، z: حدس اولیه، rho: عددی که یکی از مقادیر ویژه به آن نزدیک است.

مثال صفحه ۱۴۲

$$A = \begin{bmatrix} 2 & 3 & 2 \\ 10 & 3 & 4 \\ 3 & 6 & 1 \end{bmatrix}, z^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \rho = 0$$

```
>> A=[2 3 2;10 3 4;3 6 1];
```

```
>> [D,V]=powerInv(A,[1;1;1],0,30)
```

```
Marhaleye [1]
```

```
y =
```

```
-0.090909090909091
```

```
0.151515151515152
```

```
0.363636363636364
```

```
d =
```

```
0.363636363636364
```

```
z =
```

```
-0.250000000000000
```

```
0.416666666666667
```

```
1.000000000000000
```

```
-----  
Marhaleye [2]
```

```
y =
```

```
0.227272727272727
```

```
0.148989898989899
```

```
-0.575757575757576
```

```
d =
```

```
0.363636363636364 -0.575757575757576
```

```
z =
```

```
-0.394736842105263
```

```
-0.258771929824561
```

```
1.000000000000000
```

```
.
```

```
.
```

```
.
```

```

-----
Marhaleye [30]
y =

    0.100000762713468
    0.199999830508118
   -0.500001271189114

z =

   -0.200001016948706
   -0.399998644068392
    1.000000000000000

-----

D =

   -1.999994915256471

V =

   -0.200001016948706
   -0.399998644068392
    1.000000000000000

```

درون یابی به وسیله چند جمله ای های لاگرانژ

$C = \text{lagran}(X, Y)$

C: بردار دربردارنده ضرایب چند جمله ای برازش یافته

برای محاسبه مقدار چند جمله ای در نقطه x_0 دستور $\text{polyval}(C, x_0)$ را اجرا می کنیم.

(مثال صفحه ۱۷۲)

x	f(x)
3.2	22
2.7	17.8
1	14.2
4.8	38.3

$x_0 = 3.0$

```
>> x=[3.2 2.7 1 4.8];
>> y=[22 17.8 14.2 38.3];
>> C=lagran(x,y)

C =

    -0.5275    6.4952   -16.1177    24.3499

>> polyval(C,3)

ans =

    20.2120
```

چند جمله‌ای مورد نظر عبارتست از

$$P(x) = -0.5275x^3 + 6.4952x^2 - 16.1177x + 24.3499 \text{ \& } P(x_0 = 3) = 20.2120$$

درون‌یابی به روش نیویل

```
[p,f]=neville(x,y,xs)
```

p: ماتریسی که جدول نیویل، عناصر آن است، f: مقدار درون‌یابی انجام شده در xs

(مثال صفحه ۱۷۵)

x	f(x)
10.10	0.17537
22.20	0.37784
32.00	0.52992
41.60	0.66393
50.50	0.63608

$x_0 = 27.5$

```
>> format long
>> x=[10.1 22.2 32 41.6 50.5];
>> y=[0.17537 0.37784 .52992 0.66393 .63608];
>> [p,f]=neville(x,y,27.5)

p =

    0.6360800000000000    0.373794603960396    0.558431973072909    0.479011706129646    0.457536499191716
    0.1753700000000000    0.445241238095238    0.460710514181851    0.461738170114355             0
    0.6639300000000000    0.455998608247423    0.462003943200347             0             0
    0.3778400000000000    0.460087346938775             0             0             0
    0.5299200000000000             0             0             0             0
```

```
f =

    0.461738170114355
```

تذکر: به اختلاف ترتیب عناصر واقع در ستون p با جدول صفحه ۱۷۵ کتاب توجه کنید. دقت شود که 0.4575 به

هیچ وجه جواب سؤال نیست!



درون‌یابی به روش تفاضل محدود

(مثال صفحه ۱۸۴)

x	f(x)
0.00	0
0.20	0.03
0.40	0.423
0.60	0.684
0.80	1.03
1.00	1.557
1.20	2.572

```
D = divDiffTable(x,y)
```

در ابتدای اجرای برنامه، اگر x_i ها متساوی الفاصله بودند، ۱ و در غیر این صورت ۰ را وارد می‌کنیم.

```
>> x=[0 .2 .4 .6 .8 1 1.2];
>> y=[0 .203 .423 0.684 1.03 1.557 2.572];
>> D = divDiffTable(x,y)
motesaviol fasele? 0 or 1 1
D =
```

```

      0      0      0      0      0      0      0
0.2030  0.2030      0      0      0      0      0
0.4230  0.2200  0.0170      0      0      0      0
0.6840  0.2610  0.0410  0.0240      0      0      0
1.0300  0.3460  0.0850  0.0440  0.0200      0      0
1.5570  0.5270  0.1810  0.0960  0.0520  0.0320      0
2.5720  1.0150  0.4880  0.3070  0.2110  0.1590  0.1270
```

تذکر: برای محاسبه $f(x)$ در نقطه داده شده، مانند مثال صفحه ۱۸۸ عمل می‌کنیم.

درون‌یابی به روش حداقل مربعات

(مثال صفحه ۲۲۴)

x	0.05	0.11	0.15	0.31	0.46	0.52	0.7	0.74	0.82	0.98	1.17
y	0.956	0.89	0.832	0.717	0.571	0.539	0.378	0.37	0.306	0.242	0.104

```
C = lspoly(X,Y,M)
```

C: بردار دربردارنده ضرایب چندجمله‌ای برازش یافته، M: درجه چندجمله‌ای مورد نظر

```
>> x=[.05 .11 .15 .31 .46 .52 .7 .74 .82 .98 1.17];
>> y=[.956 .89 .832 .717 .571 .539 .378 .37 .306 .242 .104];
>> C = lspoly(x,y,2)

C =
```

```
0.2247 -1.0180 0.9980
```

برای تعیین مقدار چندجمله‌ای برازش یافته در نقطه‌ای مانند 0.76 به صورت زیر عمل می‌کنیم.

```
>> polyval(C,0.76)
```

```
ans =
```

```
0.3540
```

انتگرال‌گیری عددی به روش ذوزنقه‌ای

```
I = trapezoid(fun,a,b,npanel)
```

fun: تابع تحت انتگرال که به صورت inline وارد می‌شود، a و b حدود انتگرال، npanel: تعداد پانل‌ها

(تعداد تقسیمات محور xها)

(مثال صفحه ۲۸۷)

$$I = \int_0^1 e^x dx, n = 4$$

```
>> f=inline('exp(x)','x');
>> I = trapezoid(f,0,1,4)
```

```
I =
```

```
1.7272
```

```
>> format long
```

```
>> I
```

```
I =
```

```
1.718281828548544
```

انتگرال گیری عددی به روش $\frac{1}{3}$ سیمپسون (سیمپسون مرکب)

```
I = simpson(fun,a,b,npanel)
```

(مثال صفحه ۲۸۷)

$$I = \int_0^1 e^x dx, n = 4$$

```
>> f=inline('exp(x)','x');
>> I = simpson(f,0,1,4)
```

```
I =
```

```
1.718284154699897
```

انتگرال گیری عددی به روش رامبرگ

```
[R,quad,err,h]=romberge(f,a,b,n,tol)
```

tol: خطای قبل اگماض، R: جدول رامبرگ، quad: مقدار انتگرال، err: خطا (تفاوت دو مقدار نهایی در

جدول)، h: کوچکترین گام استفاده شده

(مثال صفحه ۲۹۰)

$$I = \int_0^1 e^x dx, n = 1$$

```
>> f=inline('exp(x)','x');
>> [R,quad,err,h]=romberge(f,0,1,1,0.00001)
```

```
R =
```

```
1.859140914229523      0      0      0      0
1.753931092464826      1.718861151876593      0      0      0
1.727221904557517      1.718318841921747      1.718282687924758      0      0
1.720518592164302      1.718284154699897      1.718281842218440      1.718281828794530      0
1.718841128579995      1.718281974051892      1.718281828675358      1.718281828460389      1.718281828459078
```

```
quad =
```

```
1.718281828459078
```

```
err =
```

```
3.354518884890467e-010
```

```
h =
```

```
0.0625000000000000
```

حل انتگرال دوگانه بر روی ناحیه مستطیلی با روش سیمپسون

```
out = simp2(f, corners)
```

تمرین ۲۹، صفحه ۳۲۷)

$$I = \int_{0.1}^{0.7} \int_{-0.2}^{0.6} e^x \sin y \, dy \, dx$$

برای حل، فاصله x و y را به ۳۰ قسمت تقسیم می کنیم.

```
>> f=inline('exp(x).*sin(y)','x','y');
>> out = simp2(f,[0.1 0.7 -0.2 0.6])
```

```
the number of subdivisions n and m in each direction must be even
enter the number of subdivisions in x and y direction as [n m] [30 30]
```

```
subdiv =
```

```
30    30
```

```
Approximate value of the integral using Simpsons rule
```

```
out =
```

```
0.140585735695089
```

حل انتگرال سه گانه بر روی ناحیه مکعب مستطیلی با روش سیمپسون

```
out = simp3(f, corners)
```

تمرین ۲۰، صفحه ۳۲۶)

$$I = \int_0^1 \int_0^1 \int_0^1 x^3 y z^2 \, dx \, dy \, dz$$

برای حل، فاصله x و y و z را به ۳۰ قسمت تقسیم می کنیم.

```
>> f=inline('x.^3.*y.*z.^2','x','y','z');
>> out = simp3(f,[0 1 0 1 0 1])
```

```
the number of subdivisions m,n and p in each direction must be even
enter the number of subdivisions [n m p] [30 30 30]
```

```
subdiv =
```

```
30    30    30
```

```
out =
```

```
0.0416666666666667
```

```
>> format rat
>> out

out =
```

```
1/24
```

حل معادلات دیفرانسیل معمولی به روش اویلر

```
E=euler(f,a,b,ya,M)
```

f : تابعی که از رابطه $y' = f$ حاصل می‌شود و به صورت inline وارد می‌شود، a و b : مقادیر ابتدا و انتها

(x_i) ها، y_a : شرط اولیه (y_0) ، M : تعداد مراحل مورد نیاز (گام بین a و b)

مثال صفحه ۳۳۴)

$$y' = f(x, y) = -2x - y, y(0) = -1$$

```
>> f=inline('-2*x-y','x','y');
>> E=euler(f,0,0.4,-1,4)
```

```
E =
```

```
0 -1.0000000000000000
0.1000000000000000 -0.9000000000000000
0.2000000000000000 -0.8300000000000000
0.3000000000000000 -0.7870000000000000
0.4000000000000000 -0.7683000000000000
```

حل معادلات دیفرانسیل معمولی به روش هیون

```
H=heun(f,a,b,ya,M)
```

مثال صفحه ۳۳۷)

$$y' = f(x, y) = 2y, y(0) = 1$$

```
>> f=inline('2*y+0*x','x','y');
>> H=heun(f,0,0.1,1,10)
```

```
H =
```

```
0 1.0000000000000000
0.0100000000000000 1.0202000000000000
0.0200000000000000 1.0408080400000000
0.0300000000000000 1.0618323624080000
0.0400000000000000 1.083281376128642
0.0500000000000000 1.105163659926440
0.0600000000000000 1.127487965856954
0.0700000000000000 1.150263222767265
0.0800000000000000 1.173498539867163
0.0900000000000000 1.197203210372480
0.1000000000000000 1.221386715222004
```

با توجه به جدول، $y(0.05) = 1.10516$ و $y(0.1) = 1.22139$ به دست می‌آید.

حل معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه دوم

```
R=rk2(f,a,b,ya,M)
```

(مثال صفحه ۳۴۲)

$$y' = f(x, y) = xy, y(2) = 1$$

```
>> f=inline('x*y','x','y');
```

```
>> R=rk2(f,2,2.1,1,1)
```

```
R =
```

```
2.0000000000000000    1.0000000000000000
2.1000000000000000    1.2260000000000000
```

حل معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه چهارم

```
R=rk4(f,a,b,ya,M)
```

(مثال صفحه ۳۴۳)

$$y' = f(x, y) = -2x - y, y(0) = -1$$

```
>> f=inline('-2*x-y','x','y');
```

```
>> R=rk4(f,0,0.5,-1,5)
```

```
k1 =
```

```
0.1000000000000000
```

```
k2 =
```

```
0.0850000000000000
```

```
k3 =
```

```
0.0857500000000000
```

```
k4 =
```

```
0.0714250000000000
```

```

-----

k1 =

    0.0714512500000000

k2 =

    0.0578786875000000

k3 =

    0.0585573156250000

k4 =

    0.0455955184375000

.
.
.
R =

           0   -1.0000000000000000
    0.1000000000000000   -0.9145125000000000
    0.2000000000000000   -0.8561927042187500
    0.3000000000000000   -0.8224552660035330
    0.4000000000000000   -0.8109608667524720
    0.5000000000000000   -0.8195928032701400

```

حل معادلات دیفرانسیل معمولی به روش میلن سیمپسون

$M = \text{milne}(f, T, Y)$

T : برداری شامل x_i ها که اولین عضو آن a و آخرین عضو آن b می باشد، Y : بردار مقادیر محاسبه شده y_i ها که از

روش رانج کوتای مرتبه چهارم به دست آمده اند.

مثال صفحه ۳۴۳

$$y' = f(x, y) = xy, y(2) = 1$$

طبق خواسته مساله باید $y(2.1)$ ، $y(2.2)$ ، $y(2.3)$ ، $y(2.4)$ را حساب کنیم. ابتدا مقادیر $y(2.1)$

، $y(2.2)$ ، $y(2.3)$ را از روش رانج کوتای مرتبه چهارم به دست می آوریم.

```

>> f=inline('x.*y','x','y');
>> R=rk4(f,2,2.3,1,3)

```

.



```

.
R =

    2.000000000000000    1.000000000000000
    2.100000000000000    1.227521897916666
    2.200000000000000    1.521952696725866
    2.300000000000000    1.905968257867614

```

مقادیر به دست آمده را در دستور میلن - سیمپسون قرار می‌دهیم.

```
>> M=milne(f,[2,2.1,2.2,2.3,2.4],[1,1.227521897916667,1.521952696725866,1.905968257867615])
```

```

M =

    2.000000000000000    1.000000000000000
    2.100000000000000    1.227521897916667
    2.200000000000000    1.521952696725866
    2.300000000000000    1.905968257867615
    2.400000000000000    2.410856827161370

```

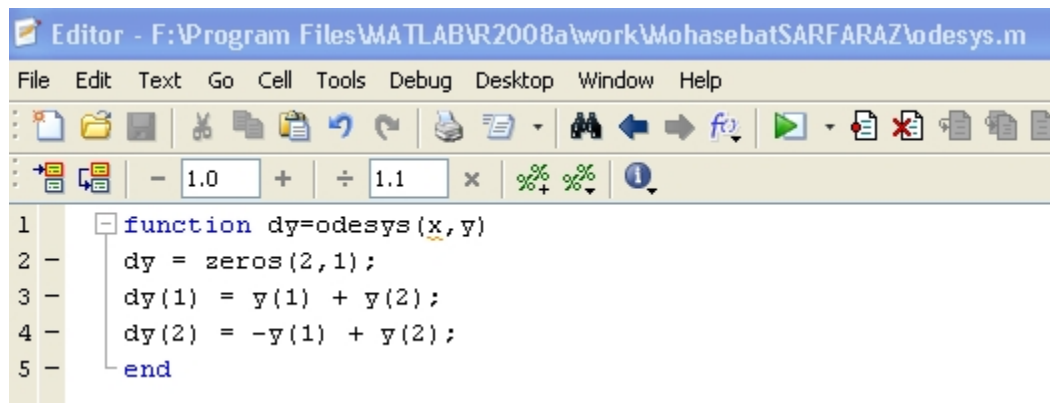
حل دستگاه معادلات دیفرانسیل معمولی به روش رانج - کوتای مرتبه چهارم

در این روش نیازی به M-file نمی‌باشد.

مثال صفحه ۳۵۱

$$\begin{cases} y_1' = y_1 + y_2 & y_1(0) = 0 \\ y_2' = -y_1 + y_2 & y_2(0) = 1 \end{cases} \quad x_s = 0.1$$

ابتدا یک M-File به شکل زیر ایجاد می‌کنیم که شامل دستگاه است.



```

Editor - F:\Program Files\MATLAB\R2008a\work\MohasebatSARFARAZ\odesys.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 function dy=odesys(x,y)
2   dy = zeros(2,1);
3   dy(1) = y(1) + y(2);
4   dy(2) = -y(1) + y(2);
5 end

```

سپس در خط فرمان، دستورات زیر را اجرا می‌کنیم.

```

>> options=odeset('Reltol',0.0001,'Abstol',[0.0001,0.0001]);
>> [X,Y]=ode45(@odesys,[0,0.1],[0,1],options)

```

```

X =

    0
    0.002500000000000000
    0.005000000000000000
    0.007500000000000000
.
.
.
    0.092500000000000000
    0.095000000000000000
    0.097500000000000000
    0.100000000000000000

Y =

    0    1.0000000000000000
    0.002506255208186    1.002499994785007
    0.005025041666358    1.004999958228854
    0.007556390624120    1.007499858846777
    0.010100333329987    1.009999664996667
    0.012656901031302    1.012499344879004
.
.
.
    0.101319834927186    1.092223755087606
    0.104310525460916    1.094700375415490
    0.107314899748192    1.097175691847733
    0.110332988730180    1.099649666829411

```

(مثال)

$$\begin{cases} y_1' = y_2 y_3, & y_1(0) = 1.1 \\ y_2' = -y_1 y_3, & y_2(0) = 2.0 \\ y_3' = -0.51 y_1 y_2, & y_3(0) = -0.7 \end{cases} \quad x_s = 6.32$$

```

>> options=odeset('Reltol',0.0001,'Abstol',[0.0001,0.0001,0.0001]);
>> [X,Y] = ode45(@odesys2,[0 6.3],[1.1 2 -0.7],options)

```

```

X =

    0
    0.024905464452960
    0.049810928905921
    0.074716393358881
    0.099621857811841
.
.

```

```

.
6.162519542033422
6.215013028022281
6.267506514011140
6.320000000000000
Y =
1.100000000000000    2.000000000000000   -0.700000000000000
1.064269715472393    2.019240035003209   -0.727625185130131
1.026817580520403    2.038540124914263   -0.754572304489877
0.987653446579336    2.057799925582575   -0.780778909411869
.
.
.
-1.434069414522162    1.775855305545289    0.241338036626603
-1.408250372405655    1.796398200841274    0.309305676325602
-1.375673106647953    1.821466343411768    0.376727440364874

```