

نرم افزار متلب

اسفند ۱۳۹۷

www.atakaraj.com

چکیده

نرم افزارها یکی از قدرتمندترین ابزارهای مهندسين هستند زیرا در دنیای مهندسی گاهی اوقات آزمون و خطا بسیار پرهزینه و زمان بر است بنابراین نیاز به محیطی مجازی داریم تا با دقت، ارزان و سریع بتوانیم به شبیه سازی و مشاهده نتایج کار خود در رایانه بپردازیم.

در این گزارش سعی بر آن است تا گام‌های ابتدایی به زبان ساده و روان شرح داده شوند و ادامه راه را برای پیمودن دانشجویان هموار نماید.

کلید واژه: نرم افزار متلب، آموزش مقدماتی متلب، ابزار مهندسی.

www.atakaraj.com

فهرست مطالب

صفحه	عنوان
ج	فهرست جدول‌ها.....
د	فهرست شکلها.....
	فهرست علائم و نشانه‌ها.....
Error! Bookmark not defined.	
فصل ۱- مقدمه ۱	
۱	۱-۱- پیشگفتار.....
	۱-۲- تاریخچه ۱.....
۱	۱-۳- هدف از گزارش.....
فصل ۲- آشنایی با محیط متلب	
۳	۲-۱- مقدمه ۳.....
۴	۲-۲- آشنایی با محیط نرم‌افزار.....
۷	۲-۳- ورود داده‌ها.....
۷	۲-۴- انواع متغیر.....
	۲-۵- m-file ها: ۸.....
۸	۲-۶- اجرای m-file ها:.....
فصل ۳- ویژگیهای اصلی نرم افزار متلب	
۹	۳-۱- مقدمه ۹.....
۹	۳-۲- ماتریس و بردار.....
۱۰	۳-۳- انواع متغیرها و مقداردهی به متغیرها در متلب.....
۱۱	۳-۴- دستورات ساده در نرم‌افزار متلب.....
۱۱	۳-۴-۱- دستور whos.....
۱۲	۳-۴-۲- دستور پاک کردن.....
	۳-۴-۳- دستور clc ۱۲.....
۱۳	۳-۵- عملگرهای ریاضی در نرم افزار متلب.....
۱۳	۳-۶- عملگرهای ریاضی مشخص شده با نماد در متلب.....
۱۴	۳-۷- عملگرهای ریاضی مشخص شده با دستور در متلب.....
۱۴	۳-۸- نمادهای پرکاربرد در متلب.....
۱۵	۳-۹- دستور فرمت:.....
۱۵	۳-۱۰- دستور plot.....
۱۶	۳-۱۱- دستور linspace.....

۱۶-۳-۱۲- دستورات مورد نیاز برای حل معادله دیفرانسیل معمولی درجه دو.....

۱۷-۱-۱۲-۳- دستور dsolve.....

۱۷-۳-۱۳- رسم جواب معادله‌ی درجه دو.....

۱۷-۳-۱۳-۱- دستور vectorize.....

۱۸-۲-۱۳-۳- دستور eval.....

۱۸-۳-۱۴- حل یک مثال.....

۲۰- فهرست مراجع.....

www.atakaraj.com

فهرست جدول‌ها

صفحه

عنوان

جدول ۱-۲	به هنگام اجرای نرم افزار چنین صفحه ای را مشاهده خواهید کرد. Error! Bookmark not defined.
جدول ۲-۲	سربرگ های اصلی نرم افزار متلب..... ۵
جدول ۳-۲	فراخوانی help در پنجره دستور..... ۶
جدول ۴-۲	جعبه ابزار فوری..... ۷
جدول ۱-۳	فراخوانی دستور whos..... Error! Bookmark not defined.
جدول ۲-۳	نحوه ی به کار بردن دستور پاک کردن..... Error! Bookmark not defined.
جدول ۳-۳	نماد عملگرهای ریاضی در متلب..... Error! Bookmark not defined.
جدول ۴-۳	عملگرهای ریاضی مشخص شده با نماد..... ۱۴
جدول ۵-۳	نمادهای پر کاربرد در متلب..... ۱۵
جدول ۶-۳	رسم جواب معادله دیفرانسیل معمولی درجه دو..... ۱۹

فهرست شکل‌ها

علامت اختصاری

عنوان

شکل ۳-۱ استفاده از تابع plot.....۱۶

www.atakaraj.com

فصل ۱ - مقدمه

۱-۱- پیشگفتار

برای انجام هر کاری نیاز به مجموعه‌ای از ابزارها نیاز داریم. یک نجار بدون اره، میخ، مداد و سایر ابزار مورد نیاز خود نمی‌تواند کاری از پیش ببرد همانطور که یک پزشک نیز بدون تجهیزات پزشکی مانند انواع وسایل جراحی و معاینه نمی‌تواند در مواقع لزوم کمک شایانی به ما کند. ما مهندسان برق نیز از این قاعده مستثنی نیستیم و ملزم به استفاده از ابزار مورد نیاز برای رسیدن به حداکثر دقت در انجام کارهای خود هستیم تا بتوانیم بهترین محصول مورد نظر خود را با کمترین هزینه و بالاترین کیفیت ارائه کنیم.

گاهی لازم است تا پیش از ساخت تجاری یک محصول نمونه آزمایشگاهی آن را مورد بررسی کامل قرار دهیم. گاهی حتی قبل از ساخت نمونه آزمایشگاهی باید آن را در یک نرم‌افزار شبیه سازی کنیم تا نقاط قوت و ضعف آن را به چشم ببینیم.

نرم افزار متلب^۱ یکی از ابزار مهندسیین برق است که یادگیری آن کمک زیادی در پیشبرد اهداف شما خواهد داشت. در این گزارش قصد داریم تا به ارائه پاره‌ای از کاربردها و بررسی این نرم افزار بپردازیم.

۱-۲- تاریخچه

متلب اواخر دهه‌ی ۱۹۷۰ توسط کلیو مولر^۲ رئیس بخش علوم کامپیوتر دانشگاه نیو مکزیکو بر پایه‌ی زبان برنامه نویسی فرترن^۳ برای اولین بار پا به عرصه‌ی حضور گذاشت. در سال ۱۹۸۳ آقای مولر با جک لیتل^۴ آشنا شد. جک لیتل متوجه پتانسیل تجاری متلب شد و همکاری خود را با مولر در زمینه‌ی بازنویسی به آغاز کرد اما اینبار از زبان برنامه نویسی C در کار خود استفاده کردند. در نهایت در سال ۱۹۸۴ ادامه راه به شرکت MathWorks سپرده شد.

۱-۳- هدف از گزارش

هدف اساسی از تهیه این گزارش معرفی اجمالی نرم‌افزار متلب به عنوان یکی از نرم‌افزارهای کاربردی در مهندسی برق است تا دانشجویان مقطع کارشناسی در ترم پنجم به گونه‌ای با یکی از مهمترین ابزار کار خود آشنا شوند و بتوانند در کمترین زمان ممکن دانش خود را در مورد نرم‌افزار متلب ارتقا دهند و برای کسب اطلاعات بیشتر در این باره ترغیب شوند. بدیهی است که به علت ژرفا و کاربردهای وسیع چنین

^۱ Matlab

^۲ Cleve Moler

^۳ Fortran

^۴ Jack Little

نرم‌افزاری توضیح کامل تمام بخش‌ها مانند کتابخانه‌ها در قالب چنین گزارشی نمی‌گنجد و از حوصله‌ی بحث خارج است اما تمام تلاش نویسنده بر این بوده است تا ضمن شیوایی بیان و استفاده از مراجع معتبر، بتواند خواننده را به تحقیق و مطالعه بیشتر در این باره ترغیب کند.

www.atakaraj.com

فصل ۲- آشنایی با محیط متلب

۲-۱- مقدمه

متلب یک سیستم جامع نرم افزاری برای محاسبات ریاضی و تکنیکی می باشد که در اصل بر پایه‌ی ماتریس^۱ این محاسبات را انجام می‌دهد. نرم افزار متلب از زبان برنامه نویسی سطح بالا استفاده می کند که شامل صدها دستور برای محاسبات ریاضی می باشد اما نباید نگران تعداد زیاد دستورها در متلب باشید زیرا معمولاً در انجام یک پروژه تنها به تعداد معدودی از آنها نیاز دارید. این نرم افزار محیطی مناسب برای انجام عملیات های ریاضی، تحلیل های آماری، بهینه سازی های ایجاد محیط های ویژوال و برنامه نویسی آن را به شکل همزمان فراهم کرده است [1].

MATLAB به معنی آزمایشگاه ماتریسی^۲ است که مفهوم بسیار مهمی توسط نام این نرم افزار به ما منتقل می شود. نخست آزمایشی بودن که مفهوم یک ابزار دانشگاهی، پژوهشی و تحقیقاتی است و دومین بخش تاکید بر ماتریسی بودن می کند، به گونه ای که همه ی انواع داده ها و تمام پردازش هایی که در متلب استفاده می کنیم به شکل ماتریس هستند. یعنی ماتریس یک ساختار پایه است.

در بحث نامگذاری این نرم افزار، غیر از نام متلب، سال ارائه و دو حرف a یا b پس از سال را مشاهده خواهید کرد. طی سال دو ویرایش از متلب توسط شرکت سازنده ارائه می شود. بدین ترتیب نام ویرایش دوم این نرم افزار برای سال جاری MATLAB 2018b خواهد بود که آخرین ویرایش ارائه شده این نرم افزار تا امروز است.

کاربرد متلب صرفاً در رشته‌ی مهندسی برق خلاصه نمی شود بلکه در سایر علوم مهندسی مانند شیمی، مکانیک، عمران و... همچنین در علوم پایه مانند ریاضی نیز از این برنامه استفاده می شود.

پیاده سازی بسیار سریع پیاده سازی ایده ها در فاز پژوهشی یکی از مهم ترین قابلیت های این نرم افزار است علی الخصوص در رشته های ریاضی و محاسباتی زیرا بسیاری از مفاهیم ریاضی در این نرم افزار تعریف شده اند و ما صرفاً نیاز به استفاده از این جعبه ابزارها^۳ داریم. مفاهیمی چون ماتریس، عدد مختلط، آرایه و... بصورت پیش فرض وجود دارند.

^۱ Matix

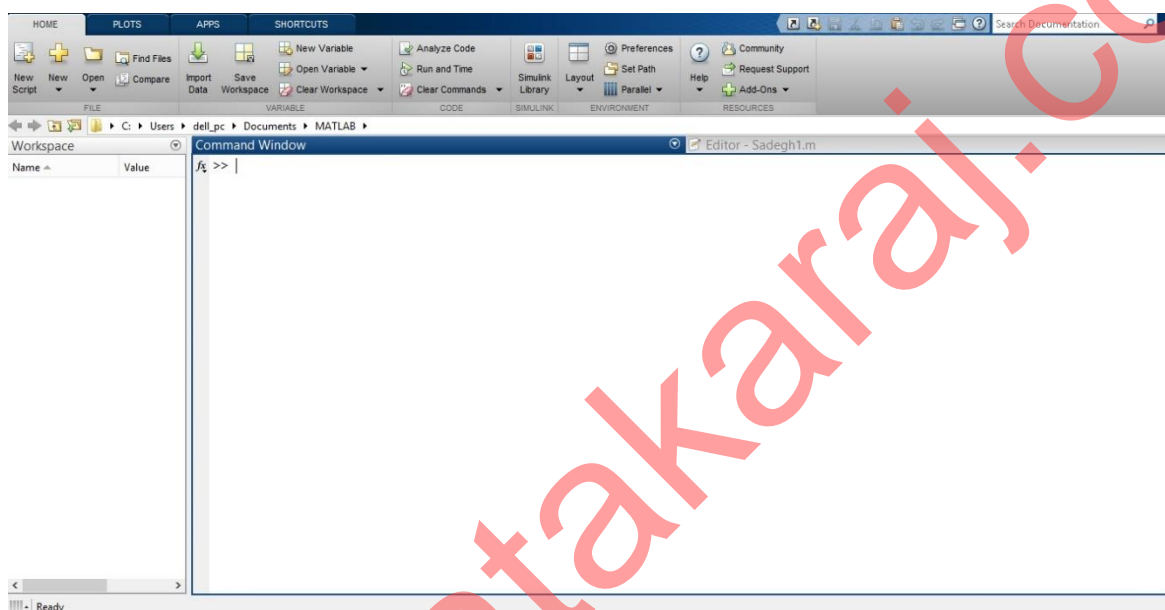
^۲ Matrix Laboratory

^۳ Tool Box

۲-۲- آشنایی با محیط نرم افزار

برای نصب نرم افزار متلب می توانید به انواع و اقسام سایتها مراجعه نمایید. حجم آخرین ویرایش این نرم افزار که MATLAB R2018b است چیزی در حدود ۱۳ گیگابایت برای سیستم عامل- ویندوز ۶۴ بیتی است. قیمت این نرم افزار نیز چیزی حدود پانزده هزار دلار است اما شما به سادگی قادر به بارگیری^۱ و نصب این نرم افزار از سایت های مختلف هستید.

پس از نصب برنامه در اولین اجرا با تصویری مانند تصویر زیر مواجه خواهید شد. البته ممکن است به علت تفاوت در نسخه^۲ نصب شده توسط شما و نسخه ی مورد استفاده توسط اینجانب (MATLAB R2014 a)، جزئیات بسیار محدود در رابط کاربری^۳ وجود داشته باشد.



جدول ۱-۲ به هنگام اجرای نرم افزار چنین صفحه ای را مشاهده خواهید کرد.

این صفحه شامل قسمت های مختلفی می باشد که به تفکیک به توضیح قسمت های مهم آن می پردازیم:

۱- پنجره دستور^۴:

همانطور که در سایر زبان های کد نویسی نیاز به نگاشتن تعدادی دستور برای رسیدن به هدفی خاص هستیم، در این نرم افزار نیز باید در پنجره ی دستور، دستورات خود را نگاشته سپس با فشردن کلید **Enter**، نتایج اجرای دستورات خود را در همین صفحه می توانیم ببینیم. این پنجره رابط نرم افزار و کاربر نیز هست. در این پنجره می توانید متغیرهایی تعریف کنید و داده ها را بکشید.

^۱ Download

^۲ Version

^۳ Interface

^۴ Command Window

۲- فضای کار^۱:

در این قسمت لیستی از متغیرهایی که به وسیله‌ی دستورات در متلب تعریف شده است، نمایش داده می‌شود.

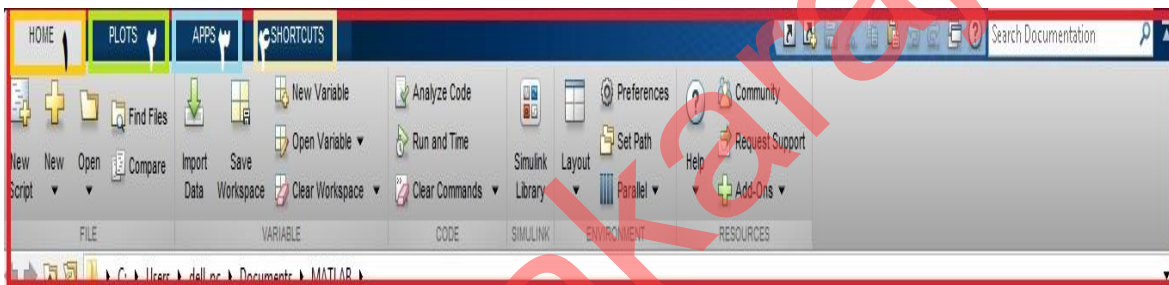
۳- پوشه‌ی فعلی^۲:

در این پنجره می‌توانیم پوشه‌ای که در آن فایل‌های متلب مورد نظرمان وجود دارد را مشاهده کنیم و به آسانی پوشه‌های مورد نظرمان را بیابیم.

۴- تاریخچه‌ی دستور^۳:

در این قسمت لیستی از دستوراتی که تا کنون توسط نرم افزار اجرا کرده‌ایم نمایش داده می‌شود.

در قسمت بالای صفحه همانطور که در شکل زیر مشاهده می‌فرمایید سه سربرگ اصلی به نام‌های HOME, PLOTS و APPS وجود دارد.



جدول ۲-۲ سربرگ‌های اصلی نرم افزار متلب

شما همواره قادر خواهید بود تا سربرگ‌های چهارم به بعد را مدیریت و شخصی سازی^۴ کنید بدین ترتیب که برخی از موارد مدنظر خود را به عنوان میانبر اضافه یا حذف نمایید. در ادامه به توضیحاتی در مورد هر کدام از این سه سربرگ می‌پردازیم.

۱- سربرگ HOME:

در این سربرگ شما قادر خواهید بود تا به انجام اقداماتی از قبیل تعریف یک پروژه جدید، باز کردن پوشه‌ی قبلی و کار با آن‌ها، تعریف متغیرهای جدید و کار با آن‌ها، فراخوانی کتابخانه‌های شبیه سازی، تغییرات و تنظیمات مربوط به رابط کاربری و استفاده از راهنمای^۵ نرم‌افزار بپردازید.

ذکر این نکته در مورد راهنمای نرم‌افزار الزامی است که این راهنما بسیار کامل و جامع می‌باشد. در مورد تمام نکات این نرم افزار و حتی مواردی از قبیل قالب^۶ نوشتن کدها می‌توانید در صورت نیاز از آن استفاده کنید. علاوه بر کلیک روی قسمت Help می‌توانید مانند سایر نرم‌افزارها از کلید میانبر F1 نیز استفاده

^۱ Workspace

^۲ Current Folder

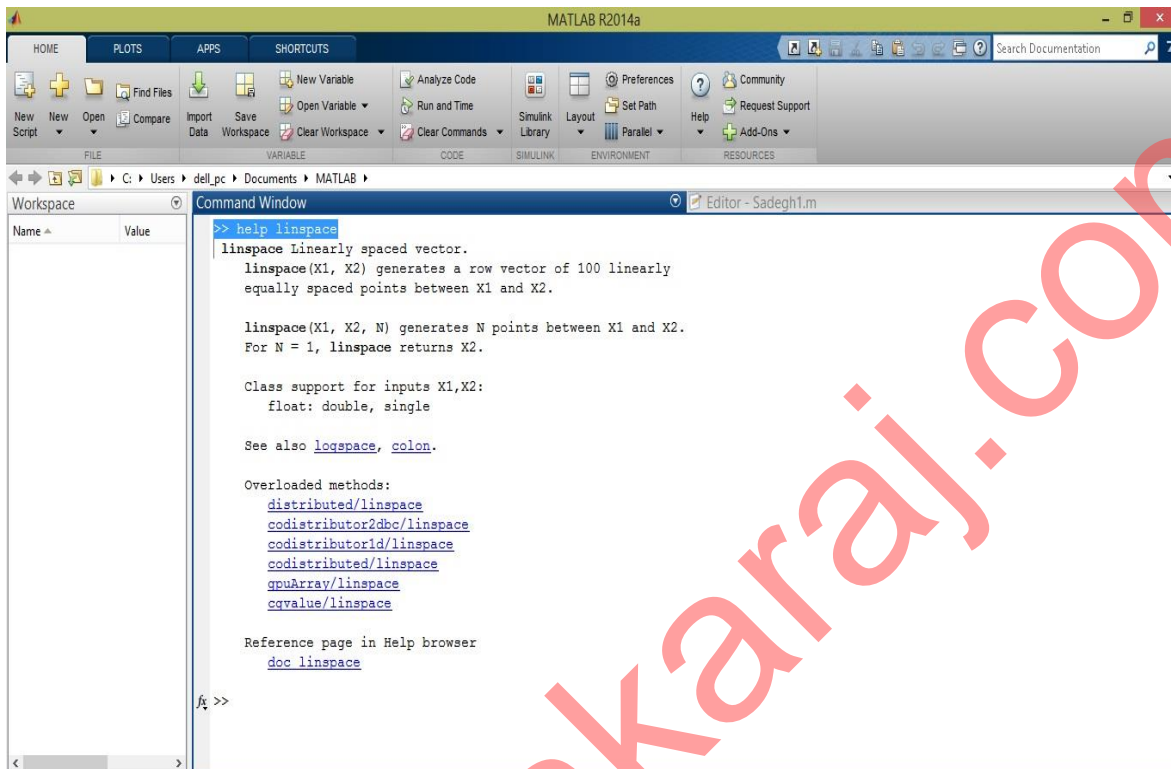
^۳ Command History

^۴ Customized

^۵ Help

^۶ Format

کنید. راه دیگری که برای استفاده از این بخش در نرم افزار پیش روی شماست، ابتدا تایپ کردن واژه‌ی help در پنجره‌ی دستور و پس از آن دستوری که می‌خواهید اطلاعاتی از آن را بدست آورید، است. مثلا در قسمت زیر در مورد linspace، نرم افزار چنین اطلاعاتی را به ما می‌دهد:



جدول ۲-۳ فراخوانی help در پنجره دستور

۲- سربرگ PLOTS:

در این قسمت همه‌ی انواع نمودارهایی که با نرم افزار متلب قابل پیاده سازی است وجود دارند که البته در ابتدا تمام آن‌ها را به رنگ خاکستری خواهید دید و به محض تعریف متغیر می‌توانید بدون حتی یک خط کد نویسی نمودار مورد نظر خود را ترسیم کنید.

۳- سربرگ APPS:

در این قسمت در واقع همان جعبه ابزار‌های متلب قرار گرفته‌اند. جعبه ابزارها در این قسمت تقسیم‌بندی شده‌اند. انواع جعبه ابزارهایی که در این قسمت هستند عبارتند از: جعبه ابزارهای مورد علاقه^۲، جعبه ابزارهای مربوط به ریاضیات، آمار و بهینه سازی، سیستم‌های کنترلی، بینایی ماشین، زیست شناسی، مالی و... که از انواع آن‌ها می‌توانیم استفاده کنیم یا حتی خودمان APP جدید تولید کنیم.

^۱ Tool Box

^۲ Favorite

قسمت بعدی که قرار است به معرفی آن بپردازیم، نوار ابزار فوری^۱ است. این قسمت به طور کلی در دو مبحث به ما کمک می‌کند. نخست اینکه می‌توان کلیدهای پر کاربرد را به آن افزود و دوم اینکه به سادگی می‌توان در قسمت Help نرم‌افزار به جستجو پرداخت.



جدول ۲-۴ جعبه ابزار فوری

۲-۳- ورود داده‌ها

برای ورود داده‌ها به چند روش می‌توانید اقدام کنید. روش اول این است که مستقیماً در پنجره‌ی دستور داده‌های موردنظر خود را تایپ کنید، روش دوم از طریق ویرایشگر^۲ در یک m-file است. به عنوان روش سوم می‌توان مقادیر را در قالب صفحه گسترده^۳ از طریق پنجره‌ی فضای کار وارد کرد.

نام متغیرها می‌تواند ترکیبی از حروف و اعداد باشد اما باید با یک حرف شروع گردد. نام متغیرها به بزرگ و کوچکی حروف حساس است. یک کتابخانه‌ی کلمات رزرو شده وجود دارد که نمی‌توان از این کلمات به عنوان نام متغیرها استفاده نمود. وقتی از این کلمات در m-file استفاده می‌شود به رنگ آبی در می‌آیند.

۲-۴- انواع متغیر

برای مثال،

$A=3;$	متغیر ساده ^۴
$A= [1\ 2\ 3\ 4];$	بردار سطری
$A = [1\ 2\ 3; 4\ 5\ 6];$	ماتریس ۳*۲

پس از ورود یک متغیر، در فضای کار نمایش داده خواهد شد. اگر روی متغیر در فضای کار دو بار کلیک شود، مقدار متغیر در یک صفحه جدید نشان داده خواهد شد.

توجه: نقطه ویرگولی که در انتهای یک عبارت قرار داده شده است، باعث متوقف کردن نمایش خروجی روی صفحه خواهد شد. گاهی اوقات برای اینکه نگاهی به داده‌های خود بیندازید مفید است، اما اگر

^۱ Quick Toolbar

^۲ Editor

^۳ Spreadsheet

^۴ Simple

تعداد زیادی داده روی صفحه نمایش نشان داده شود در حالیکه برنامه در حال اجرا است، ممکن است باعث کاهش سرعت شود.

۲-۵- m-file ها:

وقتی کد در متلب نوشته می‌شود، بهتر است که کد در عملگر ویرایشگر برنامه نوشته شود. شما می‌توانید ویرایشگر یا همان Editor را با باز کردن یک m-file موجود باز کنید. m-file قالب فایلی است که متلب می‌سازد. نگاشتن در یک m-file به شما این اجازه را خواهد داد تا کدهای خود را بسیار موثرتر ذخیره و رفع عیب کنید، نیز کامنت گذاری در این بخش بسیار کمک کننده خواهد بود.

برای گذاشتن کامنت در یک خط باید از علامت "%" در ابتدای خط استفاده کرد، مانند زیر:

% A comment line starts with a percent-sign.

۲-۶- اجرای m-file ها:

شما می‌توانید کدهای نوشته شده در متلب را به چند روش اجرا کنید. یک راه این است که m-file که از قبل نوشتید را از محلی که ذخیره کرده‌اید فراخوانی کنید. برای اینکار از پنجره‌ی دستور، دستور زیر را تایپ کنید:

```
>> run tutorial
```

tutorial نام m-file ای است که از قبل آنرا نگاشته ایم.

فصل ۳ - ویژگی‌های اصلی نرم افزار متلب

۳-۱- مقدمه

در دو فصل پیشین به مقدماتی ابتدایی در مورد نحوه‌ی کار این نرم‌افزار به همراه معرفی محیط اصلی نرم‌افزار متلب ارائه شد. کاربرد نرم‌افزار متلب همانطور که گفته شد در بسیاری از علوم در سطحی وسیع است. در این فصل قصد داریم به معرفی تعدادی از دستورات کاربردی در نرم‌افزار متلب بپردازیم. همچنین به دلیل اینکه پایه‌ی مفاهیم در نرم‌افزار متلب، ماتریس و بردار است، ابتدا کمی به چنین مفاهیمی خواهیم پرداخت.

۳-۲- ماتریس و بردار

در ریاضیات ماتریس A را بصورت زیر تعریف می‌کنیم:

$$A_{m \times n} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdot & a_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}$$

۳-۱ رابطه

فرض کنید ماتریس بالا حقیقی و در سائز $m \times n$ است. از دید نرم‌افزار متلب، این ماتریس، دو بعدی است. در ریاضیات ما صرفاً A را یک ماتریس می‌نامیم. این در حالیست که در نرم‌افزار متلب بعد ماتریس مسئله‌ی بسیار مهمی است و می‌تواند بیش از دو بعد را نیز برای یک ماتریس در نظر گرفت، برای مثال ماتریس سه بعدی زیر شامل سطر، ستون و طبقه است.

$$B = [b_{ijk}]$$

۳-۲ رابطه

البته تفاوت چندانی بین ماتریس دو بعدی و بیشتر در تعریف برای متلب وجود ندارد که در ادامه بیشتر به این قضیه خواهیم پرداخت.

تذکر: در علمی مانند فیزیک ممکن است به ماتریس، تانسور^۱ نیز گفته شود.

حال می‌خواهیم در نرم‌افزار متلب ساده‌ترین نوع ماتریس را تعریف کنیم. یک ماتریس یک در یک که اسکالر نیز هست، به طریق زیر عمل می‌کنیم:

$$A=1;$$

اما برای تعریف یک بردار باید به شکل زیر دستور را در صفحه‌ی دستور بنگاریم:

^۱ Tensor

$$B = [1 \ 5 \ 7]$$

توجه داشته باشید که در تعریف بردار از علامت قلاب استفاده کردیم. متلب هر چیزی بین این دو علامت باشد را به یکدیگر الحاق می کند که به اصطلاح به آن‌ها ^۱ CAT گفته می شود که به معنی الحاق کننده است همچنین برای تعریف برداری که متشکل از اعداد ۱، ۵ و ۷ است، می توانیم از فاصله بین دو عدد یا علامت ویرگول استفاده کنیم.

حال فرض کنید بخواهیم یک ماتریس مانند ماتریس C را در متلب تعریف کنیم. برای تعریف ماتریس از موارد گفته شده به صورت زیر استفاده می کنیم:

$$C_{2 \times 2} = \begin{bmatrix} 1 & 5 \\ 8 & 4 \end{bmatrix}$$

به این صورت در پنجره‌ی دستور می نویسیم:

$$C = [1 \ 5 ; 8 \ 4]$$

همانطور که دیده می شود برای رفتن به سطر بعد در متلب از علامت نقطه-ویرگول^۲ استفاده شد.

اگر به هر دلیلی نیاز به تنها یکی از درایه‌های ماتریس بالا داشتیم باید عبارت زیر را در پنجره‌ی دستور وارد کنیم:

$$A(i, j)$$

این دستور برای استفاده از درایه‌ی سطر i و ستون j است.

اگر به یک بردار از ماتریس C احتیاج داشتیم، بسته به اینکه بردار مورد نظر سطری یا ستونی است، از علامت نقل قول بجای آن درایه‌ی استفاده می کنیم، بعنوان مثال:

$$A(i, :) \quad \text{بردار } i \text{ ام}$$

سطر

$$A(:, j) \quad \text{بردار سطر } i \text{ ام}$$

۳-۳- انواع متغیرها و مقداردهی به متغیرها در متلب

یکی از ویژگی های متلب این است که احتیاجی نیست که حتما نوع متغیر را در همان ابتدای برنامه

مشخص کنیم و با مقادیری که در طول برنامه به متغیر نسبت داده می شود، نوع متغیر به صورت

^۱ Concatenator

^۲ Semicolon

خود به خود تعیین می شود.

در تعریف نام متغیرها باید دقت داشته باشید که متلب نسبت به کوچک یا بزرگ بودن حروف حساس می باشد.

در نرم افزار متلب ، انواع مختلفی از متغیرها وجود دارد که عبارتند از:

۱. متغیرهای عددی:

این متغیرها می توانند دارای مقادیر عددی باشند . به مثال زیر توجه کنید:

$$A = 2$$

فرض کنید بخواهیم به متغیر A، مقدار 2 را نسبت بدهیم . باید بنویسیم:

۲. متغیرهای رشته‌ای^۱:

چنانچه متغیری را بخواهیم به صورت یک رشته از حروف تعریف کنیم باید از علامت ' استفاده کنیم. به عنوان مثال:

```
s='this is a string'
```

نکته: دقت شود که استفاده از علامت ' برای ایجاد رشته‌ها ضروری است و چنانچه از این علامت استفاده نشود، با پیغام خطا مواجه می شویم .

۳. متغیرهای سمبلیک:

گاهی نیاز است که متغیر تنها به صورت سمبلیک (مثلا با حرف x) تعریف شود تا با آن معادلاتی را به صورت نمادین حل کنیم.

۳-۴- دستورات ساده در نرم‌افزار متلب

۳-۴-۱- دستور whos

چنانچه تعداد متغیرهایی که در متلب تعریف کرده اید از حدی بیشتر شود ، به سختی می توانید نام آنها را به یاد آورید. برای آنکه بتوانید تمامی متغیرهایی که در متلب تعریف کرده اید را به صورت فهرست وار ببینید تنها کافی است که دستور whos را اجرا کنید. لیست تمامی متغیرهای تعریف شده در متلب در خروجی نمایش داده می شود و نوع هر متغیر ، فضای اختصاص داده شده به آن و اندازه آن را می توانید مشاهده کنید.

^۱ String

Name	Size	Bytes	Class	Attributes
A	1x1	112	sym	
B	1x1	112	sym	
C	1x1	112	sym	
ans	1x1	112	sym	
x	1x1	112	sym	
y	1x1	112	sym	

جدول ۳-۱ فراخوانی دستور whos

اما باید دقت کنید که در فهرست فوق، مقادیر متغیرها نمایش داده نشده است. برای مشاهده مقدار هر متغیر تنها کافی است که نام متغیر را در پنجره‌ی دستور نوشته و سپس کلید Enter از صفحه کلید را بفشارید، سپس مقدار متغیر نمایش داده خواهد شد. علاوه بر این فضای کار، به صورت گرافیکی لیستی از تمامی متغیرهای تعریف شده در متلب را نمایش می‌دهد.

۳-۴-۲- دستور پاک کردن^۱

این دستور برای پاک کردن متغیرهای تعریف شده در متلب به کار می‌رود. این دستور را می‌توان به شیوه‌های زیر به کار برد:

Clear	تمامی متغیرهای تعریف شده در متلب را پاک می‌کند.
Clear all	تمامی متغیرهای تعریف شده در متلب را پاک می‌کند.
Clear x y	تنها متغیرهای x و y را پاک می‌کند.

جدول ۳-۲ نحوه‌ی به کار بردن دستور پاک کردن

نکته‌ی بسیار مهم: یک برنامه نویس متلب معمولاً اولین دستوری که در برنامه خود به کار می‌برد، دستور پاک کردن همه^۲ است، زیرا امکان این که متغیرهایی که قبلاً در متلب توسط برنامه‌های قبلی تعریف شده‌اند در برنامه جدید اختلال ایجاد کنند، زیاد است. این نکته در اجرای برنامه‌های طولانی و پیچیده از اهمیت زیادی برخوردار است.

۳-۴-۳- دستور clc

دستور clc در متلب برای پاک کردن اطلاعات نمایش داده شده در پنجره‌ی دستور به کار می‌رود. یعنی هر زمان که خواستید تمامی اطلاعات نمایش داده شده در پنجره‌ی دستور (البته تا همان لحظه، نه برای اطلاعاتی که در آینده نمایش داده می‌شوند)، پاک شوند تنها کافی است که دستور زیر را بنویسید:

clc

^۱ Clear

^۲ Clear all

البته روش دیگری نیز برای پاک کردن اطلاعات نمایش داده در پنجره‌ی دستور نیز وجود دارد که به صورت دستی است، برای این منظور ابتدا باید بر روی پنجره‌ی دستور کلیک راست، سپس گزینه‌ی پاک کردن پنجره دستور^۱ را انتخاب کرد.

۳-۵- عملگرهای ریاضی در نرم افزار متلب

در متلب برای استفاده از عملگرهای ریاضی، دو شیوه به کار رفته است. شیوه اول برای عملگرهای بسیار رایج است مانند جمع، تفریق، ضرب، تقسیم و ... که متلب برای هر کدام از آنها یک نماد را در نظر گرفته است یعنی شما در هنگام نوشتن دستورات تنها کافی است که از صفحه کلید نماد مربوط به آن عملگر را در خط دستورات وارد کنید (مثلا با فشردن دکمه * از صفحه کلید برای عملگر ضرب).

شیوه دوم، استفاده از تابع است. متلب برای عملگرهای ریاضی که میزان استفاده از آنها زیاد نیست، دستوراتی را ساخته است و شما باید نام آن دستورات و نحوه استفاده از آنها را بدانید (مثلا برای محاسبه‌ی ریشه‌ی دوم، دستور sqrt در نظر گرفته شده است).

در ادامه عملگرهای مربوط به هر دو شیوه را شرح خواهیم داد.

۳-۶- عملگرهای ریاضی مشخص شده با نماد در متلب

همانطور که گفتیم برای هر یک از این عملگرهای ریاضی، در متلب یک نماد خاص اختصاص یافته است. در جدول زیر نمادهای مربوط به عملگرهای ریاضی در متلب آمده است:

عملگر	نماد در متلب
جمع	+
تفریق	-
ضرب	*
تقسیم	/
به توان	^

جدول ۳-۳ نماد عملگرهای ریاضی در متلب

^۱ Clear Command Window

۳-۷- عملگرهای ریاضی مشخص شده با دستور در متلب

در متلب برای هر کدام از این عملگرهای ریاضی یک دستور در نظر گرفته شده است. برای نمونه تعدادی از این عملگرها را در جدول زیر آورده‌ایم:

عملگر	دستور در متلب
ریشه‌ی دوم (جذر)	sqrt
لگاریتم طبیعی (Ln)	log
لگاریتم در مبنای ۱۰	log10

جدول ۳-۴ عملگرهای ریاضی مشخص شده با نماد

۳-۸- نمادهای پرکاربرد در متلب

در نرم افزار متلب برای برخی مقادیر که معمولاً پرکاربرد هستند، نمادهایی منحصر به فرد در نظر گرفته شده است. به عنوان مثال برای عدد مشهور پی^۱ که در ریاضیات بسیار به کار می رود، نماد pi انتخاب شده است. به مثال زیر توجه کنید:

```
>> A=pi
```

```
A =
```

```
3.1416
```

برخی از نمادهای پرکاربرد در نرم افزار متلب را در جدول زیر نمایش داده‌ایم:

بی نهایت	Inf
عدد موهومی	i
کوچکترین عدد ممکن (اپسیلون) (2.2204e-016)	eps
به صورت یک عدد تعریف نشده است (مبهم) (در حالت $\frac{0}{0}$ رخ می دهد)	NaN
بزرگترین عدد حقیقی مثبت (1.7977e+308)	realmax

^۱ π

کوچکترین عدد حقیقی مثبت (2.2251e-308)	realmin
--	---------

جدول ۳-۵ نمادهای پرکاربرد در متلب

۳-۹- دستور فرمت^۱:

به صورت پیش فرض در متلب اعداد تا 4 رقم اعشار نمایش داده می شوند. اگر بخواهیم به فرمت طولانی تر تبدیل کنیم، که تعداد ارقام بیشتری را نشان دهد:

Format long;

تا ۱۵ رقم اعشار نمایش می دهد.

اگر بخواهیم به حالت قبلی بازگردانیم به طریق زیر عمل می کنیم:

Format short;

همچنین دستورات بسیاری که از حوصله ی بحث خارج است.

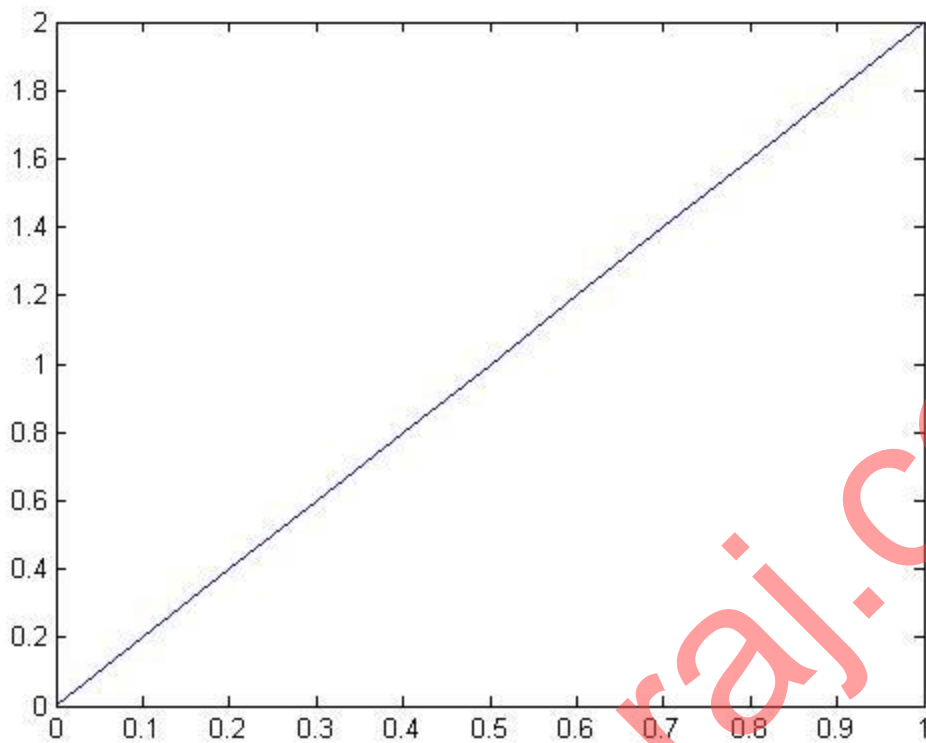
در ادامه به معرفی و کدنویسی یک دستگاه حل معادله دیفرانسیل و رسم آن در متلب می پردازیم. برای حل معادله دیفرانسیل و رسم تابع نیاز به دانستن تعدادی از دستورها داریم که معرفی خواهد شد.

۳-۱۰- دستور plot

برای رسم نمودار در متلب روش های بسیار زیادی وجود دارد. انتخاب هر روش بستگی به نوع اطلاعاتی که میخواهید نمایش داده شود دارد. گاهی اوقات فقط میخواهید اطلاعات نمایش داده شده به صورت یک نمودار ساده ی نقطه ای باشد. گاهی برای رسم نمودار در متلب احتیاج به نمودار های میله ای دارید. و گاهی نیز نمایش سه بعدی یک صفحه را میخواهید. در این گفتار، میخواهیم به ساده ترین روش برای رسم نمودار پردازیم که استفاده از دستور plot است. بردارهایی از اعداد را دریافت کرده و آنها را به صورت شکل ترسیم می کند. در واقع دستور plot مقادیر گسسته را که هر کدام به صورت یک نقطه هستند، پشت سرهم قرار می دهد و سپس آنها را با خط به هم وصل می کند تا بتوانیم آنها را به صورت یک شکل پیوسته ببینیم و بدین ترتیب به ارتباط کلی آنها پی ببریم. به طور کلی دستور plot (x,y) مقادیر y را بر حسب x رسم می کند. مثال:

```
x=0:0.1:1;
y=2*x;
plot(x,y)
```

^۱ Format



جدول ۳-۶ استفاده از تابع plot

تذکر: عدد ۰,۱ بین ۰ و ۱ را گام می‌نامند. گام به فاصله‌ای که بین دو عدد تعیین می‌شود تا متلب در آن فواصل مقدار دهی را انجام دهد و نمودار را رسم کند.

۳-۱۱- دستور linspace

دستور linspace در متلب، برای ساخت یک بردار با مقادیر عددی دارای فواصل یکنواخت (اختلاف یکنواخت) به کار می‌رود.

شکل کلی استفاده از دستور linspace به صورت زیر می‌باشد:

$A = \text{linspace}(\text{تعداد اعداد}, \text{عدد آخر}, \text{عدد اول})$

همان طور که در کد بالا نمایش داده ایم، ابتدا باید دو عدد اول و آخر را بنویسیم (یعنی کوچکترین و بزرگترین اعداد مورد نظرمان)، سپس تعداد اعداد ذکر می‌شود تا بر اساس تعداد اعداد، دستور linspace تشخیص بدهد که میزان فاصله (اختلاف) بین اعداد پشت سرهم به چه اندازه باشد.

۳-۱۲- دستورات مورد نیاز برای حل معادله دیفرانسیل معمولی درجه دو

نرم افزار متلب یک کتابخانه‌ی وسیع از توابع برای حل معادلات دیفرانسیل معمولی دارد. در این نوشتار، فقط ابتدایی‌ترین این توابع برای معادله‌ی معمولی درجه دو بررسی خواهند شد.

۳-۱۲-۱ - دستور dsolve

دستور dsolve، برای حل معادله دیفرانسیلی در متلب، به کار می رود. فرض کنید y تابعی از متغیر x باشد. معادله دیفرانسیلی، شامل مشتق مرتبه اول و یا مرتبه های بالاتر از y خواهد بود. اما چگونه باید y' را برای دستور dsolve مشخص کنیم؟ روش مورد استفاده این است که به جای y' ، از نماد D استفاده می شود و باید کل معادله در بین دو علامت کوتیشن قرار گیرد. اگر مشتق مرتبه دوم و یا بالاتر باشد، آنگاه باید ابتدا نماد D را نوشته، سپس عدد مربوط به مرتبه مشتق را بنویسیم و در آخر نیز نماد y نوشته شود. مثلاً برای تعریف y'' ، باید نماد $D2y$ و برای تعریف y''' باید نماد $D3y$ را به کار ببریم. نرم افزار متلب به طور پیش فرض متغیر t را متغیر مستقل در نظر می گیرد البته می توان متغیر مورد نظر خود را جایگزین متغیر پیش فرض کنیم و محدودیتی در این زمینه وجود ندارد.

برای تعریف شرایط اولیه از دو روش می توان استفاده کرد. مثلاً اگر شرایط اولیه به صورت زیر داشته باشیم:

$$y(1) = 1$$

روش اول این است که داخل دستور dsolve آن را تعریف کنیم. به صورت زیر:

```
dsolve ('...','y(1)=1')
```

به عنوان روش دوم نیز می توانیم یک متغیر جدید مانند زیر تعریف کنیم:

```
>> inits = 'y(1)=1';
```

۳-۱۳-۲ - رسم جواب معادله‌ی درجه دو

تا اینجا در مورد حل معادله‌ی دیفرانسیل درجه دو صحبت کردیم. فرض می شود هدف رسم جواب معادله دیفرانسیل در یک بازه‌ی مشخص باشد تا از این طریق یک دید کلی از نحوه رفتار جواب معادله دیفرانسیل حاصل شود. کاربر نرم افزار برای رسم جواب معادله دیفرانسیل با دو مشکل مواجه می شود:

۱. جواب معادله‌ی دیفرانسیل یعنی $y(x)$ نمی تواند با عملگرهای آرایه‌ای نظیر $*$ ، $.$ و $^{\wedge}$ وفق داده شود.

۲. تابع $y(x)$ که متلب به ما برمی گرداند، در اصل یک متغیر سمبولیک است.

برای حل این دو مشکل باید از دو تابع استفاده کنیم. برای حل مشکل اول از تابع $vectorize()$ و برای حل مشکل بعدی از تابع $eval()$ استفاده می کنیم.

۳-۱۳-۱ - دستور vectorize

این دستور می تواند متغیرهای سمبولیک را به رشته‌های حرفی تبدیل کند. رشته‌های حرفی می توانند با عملگرهای آرایه‌ای وفق داده شوند.

۳-۱۳-۲ - دستور eval

این دستور رشته‌های حرفی را ارزیابی می‌کند. اگر یک عبارت قانونی متلب را به صورت رشته در داخل دو آپوستروف بنویسیم نوشته‌ی ماکرو^۱ نام می‌گیرد.

تابع eval() صحت و سقم چنین عبارتی را تعیین می‌کند. در صورت صحت، تابع eval() مقدار عبارت مورد آزمایش را برمی‌گرداند، وگرنه پیغام خطای مربوطه را چاپ می‌کند. می‌توان آرگومان دومی برای eval() قرار داد که به جای پیغام خطا چاپ شود. هر دو آرگومان eval() باید به صورت رشته باشند.

۳-۱۴-۲ - حل یک مثال

در این بخش هدف حل و رسم جواب در یک بازه‌ی مشخص در معادله دیفرانسیل مرتبه دوم با کمک آموخته‌هایمان تا اینجا است. معادله‌ی دیفرانسیل معمولی درجه دو زیر را در نظر بگیرید:

$$y''_{(x)} + 8y'_{(x)} + 2y_{(x)} = \cos(x);$$

$$y_{(0)} = 0;$$

$$y_{(1)} = 1;$$

با توجه به توضیحات، کد زیر را می‌نویسیم:

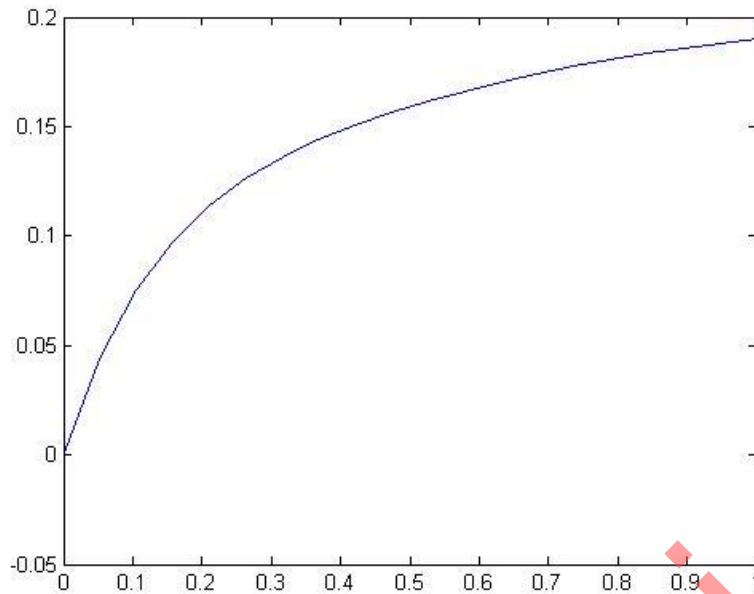
```
function Differential_system
clc
clear all
%ODE
%Solve and Plot
x=linspace(0,1,20);
eqn= 'D2y+8*Dy+2*y=cos(x)';
inits= 'y(0)=0,Dy(0)=1';
y = dsolve(eqn,inits,'x')
z = eval(vectorize(y));
plot(x,z)
```

پس از فشردن کلید اجرا^۲ یا F5 نتیجه‌ی حل معادله به صورت مجموع جواب عمومی و خصوصی در پنجره‌ی دستور و نتیجه‌ی رسم در صفحه‌ای جدید به نام شکل^۳ نشان داده خواهد شد.

^۱ Text macro

^۲ Run

^۳ Figure



جدول ۳-۷ رسم جواب معادله دیفرانسیل معمولی درجه دو

جواب حل نیز در پنجره‌ی دستور به شکل زیر نمایش داده خواهد شد:

y =

$$14^{(1/2)} \cdot \exp(4 \cdot x - 14^{(1/2)} \cdot x) \cdot \exp(x \cdot (14^{(1/2)} - 4)) \cdot (\sin(x) - \cos(x) \cdot (14^{(1/2)} - 4)) / (28 \cdot ((14^{(1/2)} - 4)^2 + 1)) - (14^{(1/2)} \cdot \exp(4 \cdot x + 14^{(1/2)} \cdot x) \cdot \exp(-x \cdot (14^{(1/2)} + 4)) \cdot (\sin(x) + \cos(x) \cdot (14^{(1/2)} + 4)) / (28 \cdot ((14^{(1/2)} + 4)^2 + 1)) - (14^{(1/2)} \cdot \exp(-x \cdot (14^{(1/2)} + 4)) \cdot (7 \cdot 14^{(1/2)} + 27)) / (28 \cdot (8 \cdot 14^{(1/2)} + 31)) - (14^{(1/2)} \cdot \exp(x \cdot (14^{(1/2)} - 4)) \cdot (393 \cdot 14^{(1/2)} + 1531)) / (28 \cdot (8 \cdot 14^{(1/2)} - 31) \cdot (8 \cdot 14^{(1/2)} + 31)^2)$$

اگر متلب نتواند یک معادله‌ی دیفرانسیل معمولی را حل کند پیغام خطای زیر را نشان می‌دهد:

Warning: Explicit solution could not be found.

www.atakaraj.com

www.atakaraj.com