اصول کدنویسی و شبیه‌سازی با MATLAB برای مهندسی برق و رباتیک

مؤلفان:
غلامرضا نظم‌آراء
نفیسه واعظی

تلفن: ۲-۶۶۴۱۹۱۰۶

سرشناسه: نظم‌آراء، غلامرضا، ۱۳۶۴-
عنوان و نام‌پدیدآور: اصول کدنویسی و شبیه‌سازی با MATLAB برای مهندسی برق و رباتیک / مولفان: غلامرضا نظم‌آراء، نفیسه واعظی
مشخصات نشر: تهران، پارسیا ۱۳۹۷
مشخصات ظاهری: ۲۶۰ ص
شابک: ۹۷۸-۶۰۰-۹۸۳۷۵-۶-۴
وضعیت فهرست‌نویسی: فیپا مختصر
یادداشت: فهرستنویسی کامل این اثر در نشانی http:// opac.nlai.ir قابل دسترسی است.
شناسه افزوده: واعظی، نفیسه، ۱۳۶۵-
شماره کتابشناسی ملی: ۵۷۲۳۵۴۰

اصول کدنویسی و شبیه‌سازی با MATLAB برای مهندسی برق و رباتیک

مؤلفان: غلامرضا نظم‌آراء، نفیسه واعظی
ناشر: پارسیا
شمارگان: ۱۰۰۰ نسخه
نوبت چاپ: اول ـ ۱۳۹۷
شابک: ۹۷۸-۶۰۰-۹۸۳۷۵-۶-۴
قیمت: ۲۲۰۰۰ تومان

مرکز پخش:

تهران، خیابان انقلاب، خیابان فخررازی، خیابان شهدای ژاندارمری نرسیده به خیابان دانشگاه ساختمان ایرانیان، پلاک ۵۸، طبقه دوم، واحد ۶ تلفن: ۹۲ـ ۶۶۴۸۴۱۹۱، www. noavarpub. com

کلیه حقوق چاپ و نشر این کتاب مطابق با قانون حقوق مؤلفان و مصنفان مصوب سال ۱۳۴۸ برای ناشر محفوظ و منحصراً متعلق به نشر پارسیا می‌باشد. لذا هر گونه استفاده از کل یا قسمتی از این کتاب (از قبیل هر نوع چاپ، فتوکپی، اسکن، عکس‌برداری، نشر الکترونیکی، هر نوع انتشار به صورت اینترنتی، سی‌دی، دی‌وی‌دی، فیلم فایل صوتی یا تصویری و غیره) بدون اجازه کتبی از نشر پارسیا ممنوع بوده و شرعاً حرام است و متخلفین تحت پیگرد قانونی قرار می‌گیرند.

فـــهرست مطالب

پیشگفتار ۶

فصل اول / ماتریس‌ها، آرایه‌ها و عملیات ریاضی ۹

مقدمه ۹
تعریف ماتریس ۹
بدست آوردن ابعاد ماتریس ۱۰
دسترسی به عناصر و زیر ماتریس‌های موجود در یک ماتریس ۱۱
بدست آوردن دترمینان، معکوس و ترانهاده‌ی یک ماتریس ۱۲
محاسبه‌ی دترمینان یک ماتریس ۱۲
محاسبه معکوس یک ماتریس ۱۲
محاسبه ترانهاده‌ی یک ماتریس ۱۲
تعریف ماتریس با درایه‌های صفر و با ابعاد دلخواه ۱۳
تعریف ماتریس با درایه‌های یک و با ابعاد دلخواه ۱۴
تعریف ماتریس همانی ۱۴
جمع و تفریق و ضرب ماتریس‌ها ۱۴
محاسبه‌ی مقادیر ویژه‌ی ماتریس ۱۸
آشنایی با دستور rand ۱۸
آرایه‌های سلولی ۱۹
تبدیل یک آرایه‌ی سلولی به یک ماتریس ۲۱
دسترسی به سلول‌های یک آرایه‌ی سلولی ۲۴
انتگرال‌گیری و محاسبه‌ی مساحت محصور یک منحنی ۲۵
مشتق گرفتن ۲۵
محاسبه‌ی حد ۲۶
حل معادله‌ی یک مجهولی ۲۷
حل دستگاه n معادله و n مجهول ۲۷
حل معادلات جبری ۲۸
آشنایی با دستور find ۲۸
آشنایی با دستور isequal ۳۰

فصل دوم / آشنایی با تکنیک‌های ترسیم و نمودارهای دو و سه بعدی ۳۲

مقدمه ۳۲
آشنایی با دستور plot ۳۲
برچسب زدن به محورها ۳۳
شبکه‌بندی ترسیم ۳۳
تعیین عنوان در شکل ۳۳
تعیین رنگ و ضخامت ترسیم ۳۳
معرفی جدول علایم اختصاری ترسیم ۳۵
ترسیم توابع ریاضی ۳۷
آشنایی با دستور inline ۴۱
رسم نمودار برای توابعی که بصورت نمادین تعریف شده‌اند با دستور ezplot ۴۱
آشنایی با دستور subplot در ترسیم ۴۲
آشنایی با کارکترهای متنی ۴۵
آشنایی با کارکترهای نشانه‌گذاری متن ۴۶
آشنایی با امکانات دستور axis ۴۸
رسم تابع چند ضابطه‌ای ۴۹
نمودارها ۵۰
نمودار میله‌ای دو بعدی و سه بعدی ۵۰
ترسیمات سه بعدی ۵۴

فصل سوم / آشنایی با شروط و حلقه‌های تکرار ۵۶

مقدمه ۵۶
آشنایی با ساختار شرطی if ۵۶
معرفی جدول عملگرهای مقایسه‌ای و منطقی ۵۷
حلقه‌ی for ۵۸
حلقه‌ی while ۵۹
ساختار switch-case ۶۰

فصل چهارم / معرفی دستورات کدنویسی سیستم‌های کنترل مدرن و خطی ۶۲

مقدمه ۶۲
تابع تبدیل ۶۲
قطب‌ها و صفرهای تابع تبدیل ۶۴
ترسیم مکان قطب‌ها و صفرهای یک تابع تبدیل در صفحه مختلط ۶۴
اتصالات سری، موازی و پسخورد توابع تبدیل ۶۶
حذف عوامل مشترک در تابع تبدیل ۶۷
محاسبه‌ی پاسخ پله‌ی سیستم ۶۸
بدست آوردن نمایش فضای حالت سیستم از روی تابع تبدیل ۶۹
بدست آوردن تابع تبدیل از روی مدل فضای حالت سیستم ۷۱
طرح یک مثال از مباحث ارائه شده تاکنون ۷۲
محاسبه‌ی صفرهای معادله‌ی مشخصه ۷۳
تعریف معادله مشخصه‌ی سیستم با تابع poly ۷۵
رسم مکان هندسی ریشه‌ها ۷۶
آشنایی با دستور rlocfind ۷۷
ترسیم نمودار بود ۷۸
ترسیم نمودار نایکویست ۷۹
محاسبه‌ی حد بهره، حد فاز، فرکانس قطع بهره و فرکانس قطع فاز ۸۲
تبدیل Z ۸۴
آشنایی با دستور c2d ۸۵

فصل پنجم / آشنایی با محیط سیمولینک و ارائه‌ی مثال‌های ترکیبی حل‌شده به‌روش سیمولینک و کدنویسی ۸۷

مقدمه ۸۷
آشنایی با محیط سیمولینک ۸۷
وارد کردن بلوک به پنجره مدل ۸۹
تنظیمات بلوک‌ها در پنجره مدل ۸۹
انتقال و چرخش بلوک ۹۱
اتصالات بین بلوک‌ها در پنجره مدل ۹۱
آشنایی با برخی ابزارهای پنجره‌ی مدل ۹۲
ذخیره کردن و تنظیم Current Folder ۹۲
اجرای فایل در سیمولینک و مشاهده‌ی نتیجه ۹۳
آشنایی با برخی تنظیمات برای پیکربندی مدل سیمولینک ۹۳
آشنایی با انواع حلگرها در متلب ۹۵
تمرین بیشتر برای کار با محیط سیمولینک ۹۶

تعریف یک ماتریس و محاسبه ترانهاده‌ی آن ۱۰۰
تعریف ماتریس صفر و یک در محیط سیمولینک ۱۰۱
تولید ماتریس با توضیع تصادفی ۱۰۳
حل معادله در محیط سیمولینک ۱۰۴
استفاده از بلوک MATLAB Function ۱۰۷
ساختار شرطی if در سیمولینک ۱۰۹
حلقه‌ی تکرار for در سیمولینک ۱۱۲
ساختار Switch-case در سیمولینک ۱۱۵
تابع تبدیل و ترسیم پاسخ پله در سیمولینک ۱۱۸
ترسیم پاسخ پله از مدل فضای حالت سیستم ۱۱۹

فصل ششم / ارتباط بین دو محیط Workspace و Simulink و آشنایی با ذخیره و بازیابی اطلاعات ۱۲۴
مقدمه ۱۲۴
آشنایی با بلوک To Workspace ۱۲۴
آشنایی با بلوک From Worksapce ۱۲۸
آشنایی با ذخیره و بازیابی کردن اطلاعات توسط متلب ۱۳۰

فصل هفتم / کنترل‌کننده‌ی PID ۱۳۳
مقدمه ۱۳۳
اصول اساسی کنترل‌کننده‌ی PID (تناسبی-انتگرالی-مشتقی) ۱۳۳
کنترل‌کننده‌ی PI ۱۳۴
کنترل‌کننده‌ی PD ۱۳۴
طراحی کنترل‌کننده‌ی PID از دو روش برای موتور جریان مستقیم ۱۳۴

فصل هشتم / ارائه‌ی مثال‌های ترکیبی از محیط سیمولینک و کدنویسی ۱۴۶
مقدمه ۱۴۶
ارائه‌ی مثال‌های ترکیبی ۱۴۶
سیستم آونگ ساده ۱۴۶
شبیه‌سازی به روش اول با استفاده از مدل فضای حالت سیستم ۱۴۸
شبیه‌سازی به روش دوم با استفاده از دو بلوک MATLAB Function ۱۵۲
شبیه‌سازی به روش سوم با استفاده از یک بلوک MATLAB Function ۱۵۵
پاسخ به ورودی‌های مشخص در تابع شبکه ۱۵۶
روش اول با استفاده از کدنویسی ۱۵۷
روش دوم حل با استفاده از محیط سیمولینک ۱۵۹
سیستم جرم-فنر و دمپر ۱۶۱
روش اول با استفاده از کد نویسی و اعمال کنترل کننده ۱۶۵
روش دوم با شبیه‌سازی در محیط سیمولینک ۱۶۶
روش سوم با شبیه‌سازی در محیط سیمولینک با استفاده از بلوک State-Space ۱۶۹
روش چهارم با شبیه‌سازی در محیط سیمولینک با استفاده از بلوک‌های MATLAB Function ۱۷۴

فصل نهم / آشنایی با شبیه‌سازی مدارهای الکترونیکی ۱۷۸
مقدمه ۱۷۸
شبیه‌سازی مدارات الکترونیکی با ارائه‌ی مثال‌های طبقه‌بندی شده ۱۷۸

فصل دهم / اصول مرتبط با طراحی دیجیتال و مدار منطقی ۱۹۰
مقدمه ۱۹۰
مرور جدول گیت‌های منطقی ۱۹۰
مثال مقدماتی آشنایی با جدول گیت‌های منطقی ۱۹۱
اصول تئوری مربوط به طراحی مدار شمارنده‌ی دو دویی ترتیبی با کلاک همزمان ۱۹۳
تعریف فلیپ فلاپ ۱۹۳
تعریف ثبات ۱۹۳
تعریف شمارنده ۱۹۳
طراحی مدار شمارنده ۱۹۳

فصل یازدهم / آشنایی با کنترل‌کننده‌ی بهینه ۱۹۸
مقدمه ۱۹۸
طراحی کنترل‌کننده از نوع تنظیم‌کننده درجه دوم خطی (LQR) ۱۹۸
مثال‌های مرتبط با طراحی کنترل‌کننده‌ی بهینه ۲۰۱

فصل دوازدهم / ایجاد مدل در محیط سیمولینک به کمک کدنویسی در محیط M-فایل ۲۱۶
مقدمه ۲۱۶
ایجاد یک مدل سیمولینکی با نام دلخواه ۲۱۶
باز نمودن مدل سیمولینکی ۲۱۶
اضافه نمودن بلوک‌ها و المان‌ها در مدل با قابلیت اختصاص نام به هر یک ۲۱۶
تنظیم پارامترهای بلوک‌ها و المان‌ها با قابلیت تنظیم موقعیت و دوران ۲۱۷
اتصال بلوک‌ها به یکدیگر ۲۱۸
ذخیره‌ی مدل ۲۱۸
اجرای مدل ۲۱۸
جمع‌بندی مباحث آموخته‌شده تاکنون در این فصل با مثال‌های طبقه‌بندی شده ۲۱۸

فصل سیزدهم / تقریب مرتبه‌ی اول و کاربرد این تقریب در کدنویسی ۲۲۳
مقدمه ۲۲۳
تقریب مرتبه اول از مشتق با ارائه‌ی مثال‌های طبقه‌بندی شده از سری زمانی ۲۲۳

فصل چهاردهم / بازوی ربات دولینکی و کنترل مفصلی ۲۴۶
مقدمه ۲۴۶
تعریف سینماتیک ربات ۲۴۶
تعریف دینامیک ربات ۲۴۶
معادله‌ی دینامیکی ربات ۲۴۶
معرفی مکانیزم ربات دو لینکی و ارائه‌ی مثال کنترل فضای مفصلی بایاده‌سازی به دو روش کدنویسی و سیمولینک ۲۴۷

فراخوان مساعدت فرهنگی و علمی

خوانندهٔ فرهیخته و بزرگوار

نشر پارسیا ضمن ارج نهادن و قدردانی از اعتماد شما به کتاب‌های این انتشارات، به استحضارتان می‌رساند که همکاران این انتشارات، اعم از مؤلّفان و مترجمان و کارگروه‌های مختلف آماده‌سازی و نشر کتاب، تمامی سعی و همّت خود را برای ارائهٔ کتابی درخور و شایستهٔ شما فرهیختهٔ گرامی به‌کار بسته‌اند و تلاش کرده‌اند که اثری را ارائه نمایند که از حدّاقل‌های استاندارد یک کتاب خوب، هم از نظر محتوایی و غنای علمی و فرهنگی و هم از نظر کیفیّت شکلی و ساختاری آن، برخوردار باشد.

بااین‌وجود، علی‌رغم تمامی تلاش‌های این انتشارات برای ارائهٔ اثری با کمترین اشکال، باز هم احتمال بروز ایراد و اشکال در کار وجود دارد و هیچ اثری را نمی‌توان الزاماً مبرّا از نقص و اِشکال دانست. ازسوی‌دیگر، این انتشارت بنابه تعهّدات حرفه‌ای و اخلاقی خود و نیز بنابه اعتقاد راسخ به حقوق مسلّم خوانندگان گرامی، سعی دارد از هر طریق ممکن، به‌ویژه از طریق فراخوان به خوانندگان گرامی، ازهرگونه اشکال احتمالی کتاب‌های منتشرهٔ خود آگاه شده و آن‌ها را در چاپ‌ها و ویرایش‌های بعدی رفع نماید.

لذا دراین راستا، از شما فرهیختهٔ گرامی تقاضا داریم درصورتی‌که حین مطالعهٔ کتاب با اشکالات، نواقص و یا ایرادهای شکلی یا محتوایی در آن برخورد نمودید، اگر اصلاحات را بر روی خود کتاب انجام داده‌اید پس از اتمام مطالعه، کتاب ویرایش‌شدهٔ خود را با هزینهٔ انتشارات پارسیا، پس از هماهنگی با انتشارات، ارسال نمایید، و نیز چنانچه اصلاحات خود را بر روی برگهٔ جداگانه‌ای یادداشت نموده‌اید، لطف کرده عکس یا اسکن برگهٔ مزبور را با ذکر نام و شمارهٔ تلفن تماس خود به ایمیل انتشارات پارسیا ارسال نمایید، تا این موارد بررسی شده و در چاپ‌ها و ویرایش‌های بعدی کتاب اِعمال و اصلاح گردد و باعث هرچه پربارترشدن محتوای کتاب و ارتقاء سطح کیفی، شکلی و ساختاری آن گردد.

نشر پارسیا، ضمن ابراز امتنان از این عمل متعهّدانه و مسئولانهٔ شما خوانندهٔ فرهیخته و گرانقدر، به‌منظور تقدیر و تشکّر از این همدلی و همکاری علمی و فرهنگی، درصورتی‌که اصلاحات درست و بجا باشند، متناسب با میزان اصلاحات، به‌رسم ادب و قدرشناسی، نسخهٔ دیگری از همان کتاب و یا چاپ اصلاح‌شدهٔ آن و نیز از سایر کتب منتشرهٔ خود را به‌عنوان هدیه، به انتخاب خودتان، برایتان ارسال می‌نماید، و درصورتی‌که اصلاحات تأثیرگذار باشند در مقدّمهٔ چاپ بعدی کتاب نیز از زحمات شما تقدیر می‌شود.

همچنین نشر پارسیا و پدیدآورندگان کتاب، از هرگونه پیشنهادها، نظرات، انتقادات و راه‌کارهای شما عزیزان در راستای بهبود کتاب، و هرچه بهتر شدن سطح کیفی و علمی آن صمیمانه و مشتاقانه استقبال می‌نمایند.

پارسیا

تلفن: ۲-۶۶۴۸۴۱۹۱

www.noavarpub.com

info@noavarpub.com

پیشگفتار

خداوند سبحان را سپاس که ما را در تدوین و نگارش این اثر یاری نمود.

هدف از تدوین این کتاب، معرفی و ارائه‌ی لیستی از مهمترین و پرکاربردترین توابع و دستورات متلب و البته مرتبط با رشته‌ی مهندسی برق و رباتیک بوده است. به طوریکه هر دو محیط کدنویسی و سیمولینک بصورت موازی در پیاده‌سازی مسائل استفاده شده است. در تدوین چهارده فصل این نوشتار، تلاش شده است که با ارائه‌ی مثال‌های ساده، به فهم، تثبیت و پیاده‌سازی مسائل کمک شود. نویسندگان این اثر، تدوین کتابی که تمام توابع موجود در محیط سیمولینک و یا M-فایل را در یک جلد در برگیرد، ضروری احساس نمی‌کنند چرا که معتقدیم بهترین مرجع آموزش این توابع، اسناد موجود در Help خود متلب و متونی است که به زبان انگلیسی در این نرم‌افزار موجود است. از طرفی پرداختن به همه‌ی آن‌ها در این کتاب باعث می‌شود ناگزیر شویم تعداد مثال‌ها و پروژه‌های کاربردی را کاهش دهیم. سال‌ها تجربه در تدریس نرم‌افزار متلب به ما نشان داده است که بهترین راه یادگیری حرفه‌ای متلب، آن است که با پروژه و مثال‌های فراوان و متعدد، آموزش انجام گیرد و از حجم مطالب اضافی و صرفا تئوری محض تا حد امکان صرف نظر شود، چرا که دانشجویان به سادگی می‌توانند با تایپ نام دستور پس تایپ کلمه‌ی Help در محیط Command Window و یا راست کلیک کردن بر روی بلوک مورد نظر در محیط سیمولینک و انتخاب گزینه Help ، حجم وسیعی از این اطلاعات را در اختیار داشته باشند. ولی استفاده از آن‌ها در مثال‌ها و پروژه‌های مرتبط با تحصیل دانشجویان است که ممکن است به راحتی در این متون یافت نشود و از این سو برایشان دردسرساز باشد.

در بسیاری از کتاب‌های موجود در بازار متاسفانه چه بسا چندین پاراگراف در مورد یک دستور توضیح داده شده، ولی هیچ مثال عملی و شهودی برای خواننده ارائه نشده است. توضیحاتی که عموما ترجمه‌ی چه بسا ناقصی از Help خود متلب است!. در این نوشتار تلاش می‌شود ضمن کاهش توضیحات فرار و اضافی، مستقیما به حل مسائل بپردازیم. با این استراتژی تلاش شده است که پنجره‌ای پیش سوی یک برنامه‌نویس و مهندس مبتکر بگشاییم که با نگرش و مطالعه در این قاب، قادر باشد با حداقل اطلاعات، حداکثر بهره‌وری را سبب شود. استراتژی ما، یادگیری فنون و تکنیک‌های نرم‌افزار متلب از مثال‌ها و پروژه‌ها است. این تکنیک نه تنها به یادگیری سرعت و شتاب شگفت‌انگیزی می‌بخشد، بلکه باعث می‌شود دانشجویان با ابتکار و ذوق مهندسی، راه حل‌ها و فنون جدیدی را خود پیشنهاد دهند و گاهی اوقات خودشان مطالبی را کشف نموده و مطالب را به شیوه‌ای خودآموز فرا گیرند. بر ما ثابت شده است ماندگاری مطالب تمرین شده و فراگرفته شده در این تکنیک به مراتب بیشتر از روش‌های مرسوم است.

این اثر در چهارده فصل و مطابق با سرفصل‌های زیر تدوین شده است. در فصل اول با عملیات آرایه‌ای و محاسباتی، در فصل دوم با تکنیک‌های ترسیم، در فصل سوم با شروط و حلقه‌های تکرار، در فصل چهارم با اصول کدنویسی سیستم‌های کنترل مدرن و خطی، در فصل پنجم با محیط سیمولینک و مثال‌های ترکیبی، در فصل ششم با ارتباط بین دو محیط سیمولینک و کدنویسی و ذخیره و بازیابی اطلاعات، در فصل هفتم با کنترل‌کننده‌ی PID، در فصل هشتم با ارائه‌ی مثال‌های ترکیبی کدنویسی و سیمولینک با حل موازی و مقایسه‌ای، در فصل نهم با شبیه‌سازی مدارات الکترونیکی، در فصل دهم با اصول طراحی دیجیتال و مدار منطقی در سیمولینک، در فصل یازدهم با اصول و پیاده‌سازی کنترل‌کننده بهینه، درفصل دوازهم با اصول ایجاد مدل سیمولینکی به کمک کدنویسی، در فصل سیزدهم با اصول تقریب مرتبه اول و کاربرد آن در کدنویسی و در نهایت در فصل چهاردهم با معادلات دینامیکی ربات دو لینکی و کنترل فضای مفصلی آشنا خواهید شد.

در این اثر سعی شده از تکنیک‌ها و روش‌های متعددی برای حل مساله‌ها استفاده شود. ضمن دقت و وسواس فراوان در انتخاب مثال‌های آموزشی، گاهی اوقات یک مثال از چند روش حل شده است تا امکان مقایسه و نقد نقاط قوت و ضعف هر روش برای خواننده اثر بیشتر تداعی شود. از طرفی ترتیب پیشبرد و ارائه‌ی مطالب در این نوشتار به گونه‌ای است که گام به گام، دانشجویان را با مطالب سطح بالاتر آشنا نماید. بنابراین برای حصول حداکثر بهره‌وری و کاهش زمان یادگیری توصیه می‌کنیم که مطالعه‌ی این اثر با ترتیب ارائه‌ی مثال‌های پیشنهادی در هر فصل صورت پذیرد. در نگارش این کتاب، سعی شده است که طبقه‌بندی و انتخاب مثال‌ها و توضیحات لازم به گونه‌ای باشند که دانشجویان و خوانندگان عزیز، حضور در یک کلاس درس پر نشاط متلب را احساس کنند.

کلیه‌ی پروژه‌ها و کدنویسی‌ها در نسخه‌ی 2017b متلب پیاده‌سازی شده‌اند. در تدوین این نوشتار، سعی شده از توابع و دستوراتی که با توجه به تجربه‌ی خود، احتمال حذفشان از نسخ آینده‌ی این نرم‌افزار را می‌دادیم استفاده نکنیم و تکنیک‌های مفید و روشی مدوّن و یکپارچه برای حل مسائل ارائه دهیم. معتقدیم که چنانچه دانشجویان عزیز بر این تکنیک‌ها تسلط یابند، خواهند توانست از عهده‌ی پیاده‌سازی و شبیه‌سازی بسیاری از مسائل پیچیده‌ی مهندسی برآیند.

علی‌رغم کوشش و اهتمام فراوان نگارندگان در تصحیح و بازبینی این نوشتار، وجود کاستی و نقصان در محتوای این اثر امریست اجتناب‌ناپذیر که امید است با ارسال نظرات، پیشنهادات و رهنمودهای ارزشمندتان به آدرس الکترونیکی انتشارات پارسیا ما را یاری فرمائید.

در پایان لازم می‌دانیم مراتب سپاس و قدردانی خود را به محضر استاد ارجمندمان جناب آقای پروفسور محمد مهدی فاتح که بی‌شک آشنایی با الفبای مهندسی برق و رباتیک و نرم‌افزار متلب را مرهون زحمات بی‌دریغ این استاد بزرگوار هستیم، تقدیم نماییم. همینطور از آقای مهندس محمدرضا واعظی و همینطور آقای مهندس بهروز نظم‌آرا که در تالیف این کتاب، همواره همیار ما بودند، تشکر و قدردانی می‌کنیم.

هشـــدار

کلّیة حقوق چاپ و نشر این کتاب مطابق با قانون حقوق مؤلّفان و مصنّفان و هنرمنـدان مصوّب سال ۱۳۴۸ و آیین‌نامة اجرایی آن مصوّب ۱۳۵۰، برای ناشر محفـوظ و منحصـراً متعلّـق بـه نشر پارسیا است. لذا هر گونه استفاده از کل یا قسمتی از مطالب، اشـکال، نمودارهـا، جـداول، تصـاویر ایـن کتاب در دیگر کتب، مجلات، نشریات، سایت‌ها و موارد دیگر، و نیز هر گونه اسـتفاده از کـل یـا قسـمتی از کتاب به هر شکل از قبیل هر نـوع چـاپ، فتـوکپی، اسـکن، تایـپ از کتـاب، تهیّـة پـی دی اف از کتـاب، عکس‌برداری، نشر الکترونیکی، هر نـوع انتشـار بـه‌صـورت اینترنتـی، سـی دی، دی وی دی، فـیلم، فایـل صوتی یا تصویری و غیره بـدون اجازة کتبی از نشر پارسیا ممنوع و غیرقانونی بـوده و **شـرعاً نیـز حـرام** است، و متخلّفین تحت **پیگرد قانونی و قضایی** قرار می‌گیرند.

با توجّه به اینکه هیچ کتابی از کتب نشر پارسیا به‌صـورت فایـل ورد یـا پـی دی اف و مـوارد این‌چنین، توسط این انتشارات در هیچ سایت اینترنتی ارائه نشـده اسـت، لـذا درصـورتی‌کـه هـر سایتی اقدام به تایپ، اسکن و یا موارد مشابه نماید و کل یا قسمتی از متن کتب نشـر پارسـیا را در سایت خود قرار داده و یا اقدام به فـروش آن نمایـد، توسـط کارشناسـان امـور اینترنتـی ایـن انتشارات، که مسئولیّت ادارة سایت را به عهده دارند و به طور روزانه به بررسی محتوای سـایت‌هـا می‌پردازند، بررسی و درصورت مشخص شدن هرگونه تخلّف، ضمن اینکه این کار از نظـر قـانونی غیرمجاز و از نظر شرعی نیز حرام می‌باشد، وکیـل قـانونی انتشـارات از طریـق وزارت فرهنـگ و ارشاد اسلامی، **پلیس فتا** (پلیس رسیدگی به جـرایم رایانـه‌ای و اینترنتـی) و نیـز سـایر مراجـع قانونی، اقدام به مسدود نمودن سـایت متخلّـف کـرده و طـیّ انجـام مراحـل قـانونی و اقـدامات قضایی، خاطیان را مـورد **پیگرد قــانونی و قضــایی** قـرار داده و کلّیّـة خسـارات وارده بـه ایـن انتشارات از متخلّف اخذ می‌گردد.

همچنین درصورتی‌که هر کتابفروشی، اقدام بـه تهیّـة کپـی، جـزوه، چـاپ دیجیتـال، چـاپ ریسو، اُفست از کتب انتشارات پارسیا نموده و اقـدام بـه فـروش آن نمایـد، ضـمن اطـلاع‌رسـانی تخلّفات کتابفروشی مزبور به سایر همکاران و مُوَزّعین محترم، از طریـق وزارت فرهنـگ و ارشـاد اسلامی، اتحادیة ناشران، و انجمن ناشران دانشـگاهی و نیـز مراجـع قـانونی و قضـایی اقـدام بـه استیفای حقوق خود از کتابفروشی متخلّف می‌نماید.

خرید، فروش، تهیّه، استفاده و مطالعه از روی نسخة غیراصلِ کتاب،
از نظر قانونی غیرمجاز و شرعاً نیز حرام است.

انتشارات پارسیا از خوانندگان گرامی خود درخواست دارد که در صورت مشاهدة هـر گونـه تخلّـف از قبیـل مـوارد فـوق، مراتب را یا از طـریق تلفـن‌هـای انتشـارات پارسـیا بـه شـماره‌هـای۲- ۶۶۴۸۴۱۹۱ ۰۲۱ و ۰۹۱۲۳۰۷۶۷۴۸ و یا از طریـق ایمیـل انتشـارات بـه آدرس *info@noavarpub.com* و یـا از طریق منوی تماس با ما در سایت *www.noavarpub.com* به این انتشارات ابلاغ نمایند، تـا از تضـییع حقـوق ناشـر، پدیدآورنـده و نیـز خـود خواننـدگان محتـرم جلـوگیری بـه‌عمـل آیـد، و نیز به‌عنوان تشکّر و قدردانی، از کتب انتشارات پارسیا نیز هدیه دریافت نمایند.

فصل اول

ماتریس‌ها، آرایه‌ها و عملیات ریاضی

مقدمه

کلمه‌ی MATLAB برگرفته از عبارت " MATRIX LABORATORY " است. در این فصل مطالب مقدماتی در خصوص بردارها و ماتریس‌ها، برخی ویژگی‌های خاص آن‌ها و عملیات ماتریسی و ریاضی مرتبط با آن‌ها ارائه خواهد شد و سپس در ادامه با آرایه‌های سلولی و تعریف و چگونگی دسترسی به آن‌ها آشنا خواهیم شد. بی شک، یادگیری این مباحث، پیش‌نیازی ضروری برای پیاده‌سازی و شبیه‌سازی مسائل مهندسی تلقی شده و از آن‌ها به کرار در فصول آینده استفاده خواهیم نمود.

تعریف ماتریس

یک ماتریس متشکل از r سطر و c ستون که دارای $r \times c$ عنصر است، در حالت کلی بصورت زیر در نظر گرفته می‌شود.

$$M_{r\times c} = \begin{bmatrix} m_{11} & \cdots & m_{1c} \\ \vdots & \ddots & \vdots \\ m_{r1} & \cdots & m_{rc} \end{bmatrix}$$

توجه کنید که عنصر سطر i-ام و ستون j-ام از این ماتریس با m_{ij} نمایش داده می‌شود که هر یک از مولفه‌های ماتریس را تشکیل می‌دهند. توجه کنید که عبارت $M_{r\times c}$ بیانگر ماتریسی با ابعاد $r \times c$ است.

ماتریس زیر را در نظر بگیرید:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}$$

این ماتریس دارای ۳ سطر و ۴ ستون می‌باشد، که در مجموع دارای 3×4 عنصر می‌باشد، به دستور ساخت ماتریس فوق در محیط پنجره فرمان متلب یا محیط M-فایل توجه کنید که بصورت زیر می‌باشد،

```
A=[1 2 3 4;5 6 7 8;9 10 11 12]
```

برای جدا کردن مولفه‌ها از لحاظ ستونی می‌توان از فاصله و یا کاما ',' استفاده نمود و برای تعریف سطر جدید در متلب از دستور سمیکولن ';' استفاده می‌شود. با این توضیحات دستور فوق در متلب

می‌توانست بصورت زیر نیز نوشته شود.

```
A=[1,2,3,4;5,6,7,8;9,10,11,12]
```

در ادامه لازم است با برخی از خواص ساده‌ی ماتریس‌ها بیشتر آشنا شویم.

بدست آوردن ابعاد ماتریس

با دستور زیر ابعاد ماتریس A در محیط متلب نمایش داده می‌شود.

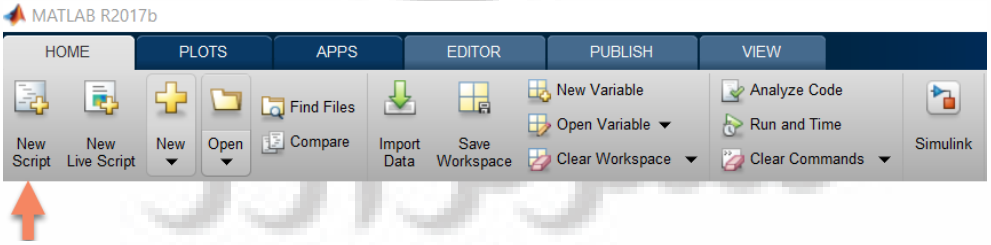
```
size(A)
```

مثلا در صورتی که ماتریس A را بصورت قبل تعریف کرده باشید، دستور فوق ابعاد ماتریس را بصورت 3×4 نمایش می‌دهد. به برنامه‌ی جدول (۱-۱)، برای آشنایی با کاربرد صحیح این دستور و مشاهده‌ی نتیجه در محیط Command Window به صورت شکل (۱-۳) توجه کنید.

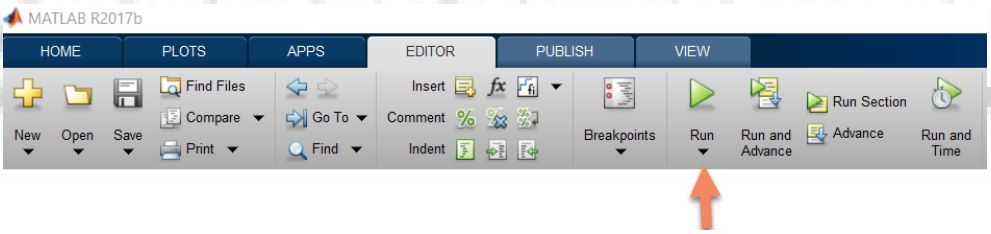
جدول ۱-۱ –برنامه‌ی مربوط به تعریف ماتریس و محاسبه ابعاد آن

```
clc;
clear all;
close all;
A=[1 2 3 4;5 6 7 8;9 10 11 12];
s=size(A)
```

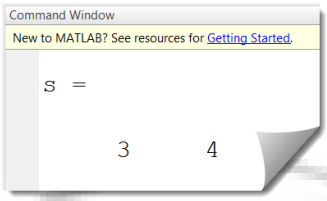
برنامه‌ی جدول (۱-۱) را در یک فایل Script تایپ نموده و سپس این برنامه را اجرا نمایید. طریقه‌ی ایجاد یک فایل Script یا یک M-فایل خالی در شکل (۱-۱) و از قسمت HOME نشان داده شده است. برای اجرای برنامه نیز بر روی Run مطابق شکل (۱-۲) از قسمت EDITOR کلیک نمایید.



شکل ۱-۱- ایجاد یک فایل *Script* یا *M*-فایل



شکل ۱-۲- کلیک بر بروی گزینه *Run* برای اجرای برنامه



شکل ۱-۳- نمایش اجرای برنامه‌ی جدول (۱-۱) در محیط *Command Window*

در پایان دستور size(A)، سمیکولن گذاشته نشده است، این امر سبب می‌شود که نتیجه در محیط Command Window نمایش داده شود. در صورتی که از سمیکولن در پایان یک دستور استفاده کنیم، مقدار متغیر در پنجره فرمان نمایش داده نمی‌شود. در اینصورت برای نمایش آن کافیست پس از تایپ نام آن در محیط پنجره‌ی فرمان، کلید Enter را بزنیم.

در اکثر برنامه‌های موجود در جداول این کتاب از سه دستور زیر در ابتدای برنامه استفاده شده است.

```
clc;
clear all;
close all;
```

این سه دستور برای اطمینان از عدم تداخل اجرای یک برنامه با برنامه‌های دیگر و اجراهای پیشین، معمولا در ابتدای یک برنامه نوشته می‌شوند. دستور 'clc' برای پاک کردن صفحه‌ی پنجره‌ی فرمان یا محیط Command Window، دستور 'clear all' برای پاک کردن متغیرها در متلب و مقدار هر یک و دستور 'close all' نیز برای بستن پنجره‌ی شکل‌های باز حاصل از اجراهای پیشین به کار می‌رود.

دسترسی به عناصر و زیر ماتریس‌های موجود در یک ماتریس

به عنوان مثال فرض کنید عنصر سطر دوم و ستون سوم ماتریس A بصورت نشان داده شده در شکل (۱-۴) را می‌خواهیم جدا کنیم،

$$A=\begin{bmatrix}1 & 2 & 3 & 4\\ 5 & 6 & 7 & 8\\ 9 & 10 & 11 & 12\end{bmatrix}$$

شکل ۱-۴

کافیست از دستور زیر استفاده کنیم.

```
A(2,3)
```

در صورتی که بخواهیم ناحیه‌ی مشخص شده بصورت نشان داده شده در شکل (۱-۵) در ماتریس A را جدا کنیم کافیست، از دستور زیر استفاده کنیم.

```
A(1:2,2:3)
```

$$A=\begin{bmatrix}1 & 2 & 3 & 4\\ 5 & 6 & 7 & 8\\ 9 & 10 & 11 & 12\end{bmatrix}$$

شکل ۱-۵

این دستور به این معنا است که عناصر موجود در سطر ۱ تا سطر ۲ و همینطور ستون ۲ تا ستون ۳ از ماتریس A جدا می‌شود و بصورت یک ماتریس جدید نمایش داده می‌شود. در صورتی که برداری شامل مولفه‌هایی از ماتریس A بصورت نشان داده شده در تصویر شکل (۱-۶) مد نظر باشد با دستور زیر این بردار از ماتریس جدا می‌شود.

```
A(2,2:4)
```

$$A=\begin{bmatrix}1 & 2 & 3 & 4\\ 5 & 6 & 7 & 8\\ 9 & 10 & 11 & 12\end{bmatrix}$$

شکل ۱-۶

توجه کنید که توسط دستور فوق، سطر دوم ماتریس A و از ستون دوم تا ستون چهارم بصورت مستقل نمایش داده می‌شود.

بدست آوردن دترمینان، معکوس و ترانهاده‌ی یک ماتریس

به یاد داریم که یک شرط لازم برای محاسبه‌ی دترمینان یک ماتریس و ماتریس معکوس آن این بود که ماتریس مربعی باشد، یعنی تعداد سطرهای ماتریس با تعداد ستون‌های آن برابر باشد. ماتریس مربعی زیر را در نظر بگیرید:

$$B=\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}$$

محاسبه‌ی دترمینان یک ماتریس

دترمینان این ماتریس با دستور زیر محاسبه می‌گردد.

```
det(B)
```

محاسبه معکوس یک ماتریس

از طرفی ماتریس معکوس با دستور زیر قابل محاسبه است.

```
B^-1
```

می‌توان معکوس ماتریس را از دستور زیر نیز محاسبه نمود.

```
inv(B)
```

محاسبه ترانهاده‌ی یک ماتریس

ترانهاده‌ی این ماتریس با دستور 'B محاسبه می‌گردد.

```
B'
```

از طرفی می‌توانیم ترانهاده‌ی ماتریس دلخواه B را با دستور $transpose(B)$ نیز بدست آوریم.

```
transpose(B)
```

برای آشنایی بیشتر با دستوراتی که تاکنون معرفی شده، به مثال (۱-۱)، توجه کنید.

مثال ۱-۱: ماتریس $A=\begin{bmatrix}1 & 2 & 3\\0 & 2 & 0\\3 & 2 & 1\end{bmatrix}$ موجود است. برنامه‌ای بنویسید که دترمینان ، معکوس و ترانهاده‌ی این ماتریس را محاسبه نموده و به ترتیب در متغیرهای B ، C و D قرار دهد.

با اجرای برنامه‌ی جدول (۱-۲)، مقادیر خواسته شده پس از محاسبه توسط دستورات متلب، به ترتیب در متغیرهای مورد نظر قرار گرفته و بدلیل اینکه در پایان دستورات علامت سمیکولن گذاشته نشده، نتیجه در محیط Command Window نمایش داده می‌شود.

جدول ۱-۲ –برنامه‌ی مربوط به مثال (۱-۱)

```
clc;
clear all;
close all;
A=[1 2 3;0 2 0;3 2 1];
B=det(A)
C=inv(A)
D=transpose(A)
```

از طرفی می‌توانستیم برنامه را بصورت جدول (۱-۳)، نیز پیشنهاد دهیم که نتیجه کاملا مشابهی حاصل می‌شد.

جدول ۱-۳ –برنامه مربوط به مثال (۱-۱)

```
clc;
clear all;
close all;
A=[1,2,3;0,2,0;3,2,1];
B=det(A)
C=A^-1
D=A'
```

تعریف ماتریس با درایه‌های صفر و با ابعاد دلخواه

دستور $zeros\left(\text{تعداد ستون ,تعداد سطر}\right)$، ماتریس صفر با تعداد سطر و ستون دلخواه را ایجاد می‌کند. به عنوان مثال با دستور زیر:

```
zeros(2,3)
```

ماتریس صفری بصورت $\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\end{bmatrix}$ تعریف شده و در محیط پنجره‌ی فرمان متلب نمایش داده می‌شود که دارای دو سطر و سه ستون صفر می‌باشد.

تعریف ماتریس با درایه‌های یک و با ابعاد دلخواه

دستور (تعداد ستون ,تعداد سطر) *ones*، ماتریس با درایه‌های یک و با تعداد سطر و ستون دلخواه را ایجاد می‌کند. به عنوان مثال با دستور زیر:

```
ones(3,2)
```

ماتریسی بصورت $\begin{bmatrix}1 & 1\\1 & 1\\1 & 1\end{bmatrix}$ در محیط پنجره‌ی فرمان متلب نمایش داده می‌شود که دارای سه سطر و دو ستون با درایه‌های یک می‌باشد.

تعریف ماتریس همانی

ماتریس همانی، یعنی ماتریسی که درایه‌های موجود در روی قطر اصلی آن یک باشد و دیگر درایه‌های آن صفر باشد. به عنوان مثال دستور $eye(3,3)$ ماتریس همانی بصورت $\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$ تولید می‌کند. از طرفی اگر دستور eye مربعی تعریف نشود، درایه‌های موجود در قطر اصلی بزرگترین ماتریس مربعی موجود از سمت چپ داخل آن یک می‌شود و بقیه صفر می‌شود. به عنوان مثال دستور $eye(2,4)$ نتیجه‌ای بصورت $\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\end{bmatrix}$ در محیط پنجره فرمان متلب نمایش می‌دهد. توجه کنید که برای ماتریس‌های مربعی می‌توانیم تنها از یک عدد در آرگومان این دستورات نیز استفاده کنیم. به عنوان مثال، دستورات $zeros(2)$، $ones(2)$ و $eye(3)$ به ترتیب ماتریس‌های $\begin{bmatrix}0 & 0\\0 & 0\end{bmatrix}$، $\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}$ و $\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$ را تعریف می‌کنند. متلب عدد موجود در داخل آرگومان این دستورات را به سطر و ستون هر یک نسبت می‌دهد.

جمع و تفریق و ضرب ماتریس‌ها

با سه دستور زیر به ترتیب عملیات جمع، تفریق و ضرب دو ماتریس صورت می‌پذیرد.

```
A+B
A-B
A*B
```

برای آشنایی بیشتر با عملیات جمع، تفریق و ضرب ماتریس‌ها به مثال (۱-۲) توجه کنید.

تلفن: ۶۶۴۸۲۱۹۱-۲۰

مثال ۱-۲: برنامه‌ای بنویسید که ماتریس M را بصورت زیر تعریف کند و سپس درایه‌های مشخص شده در آن را به ترتیب در ماتریس‌های A و B قرار دهد و ابعاد ماتریس M را نیز در متغیر s ذخیره نماید و سپس حاصل عبارت‌های $C_1 = A + B$ ، $C_2 = A - B$ ، $C_3 = AB$ ، $C_4 = (AB)^{-1}$ و $c_5 = |AB|$ را محاسبه نماید.

$$M = \begin{bmatrix} \overbrace{\begin{matrix}1 & -1\\ 4 & 1\end{matrix}}^{A} & 0 & 2 & 0 \\ 2 & 9 & 2 & \\ -2 & 0 & 3 & \overbrace{\begin{matrix}1 & 5\\ 1 & 3\end{matrix}}^{B} \\ 0 & 2 & 0 & \end{bmatrix}$$

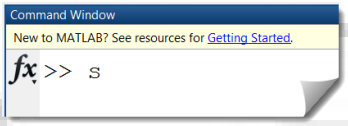
در داخل یک M-فایل، برنامه‌ی جدول (۱-۴) را تایپ نمایید و سپس در مسیر داخواه ذخیره نمایید و در پایان نیز برنامه را Run نمایید. توجه کنید که پسوند یک فایل M-فایل بصورت 'm.' است.

جدول ۱-۴ –برنامه تعریف ماتریس M انجام عملیات بر روی آن مربوط به مثال (۱-۲)

```
clc;
clear all;
close all;
%% program
M=[1 -1 0 2 0;4 1 2 9 2;-2 0 3 1 5;0 2 0 1 3];
A=M(1:2,1:2);B=M(3:4,4:5);
s=size(M);
C_1=A+B
C_2=A-B
C_3=A*B
C_4=inv(A*B)% you can also type C_4=(A*B)^-1
c_5=det(A*B)
```

توضیحات اضافی در برنامه می‌تواند بعد از علامت ٪ آورده شود، محیط متلب خطوطی که با این علامت شروع شوند را نادیده می‌گیرد و اجرا نمی‌کند. چنانچه از دو علامت %% در ابتدای خط برنامه استفاده شود نواحی زیرین این دستور تا علامت %% بعدی با تغییر رنگ پس زمینه از هم مجزا می‌شوند در این صورت توضیحات اضافی که پس از %% نوشته شده‌اند، بصورت برجسته یا Bold نمایش داده می‌شوند. از طرفی توجه کنید که اگر در پایان دستورات از ';' یا سمیکولن استفاده نشود، خروجی در صفحه دستور متلب نمایش داده می‌شود، چنانچه سمیکولن در پایان دستوری قرار دهید، برای نمایش آن در محیط Command Window باید پس از تایپ نام آن در مقابل خط فرمان کلید Enter بزنید. با اجرای برنامه‌ی جدول (۱-۴) همانطور که مشاهده می‌کنید متغیرهای s ، A و B در

محیط پنجره فرمان متلب نمایش داده نمی‌شوند. در این صورت پس از اجرای برنامه M –فایل، باید در محیط Command Window برای نمایش بردار s ، عبارت s را در مقابل خط فرمان یا Command Line بصورت شکل (۱-۷) تایپ نمایید.



شکل ۱-۷

سپس Enter بزنید تا نتیجه قابل نمایش باشد. در صورت نیاز بـه مشـاهده‌ی مـاتریس‌هـای A و B عملیات مشابه بردار s را تکرار می‌کنیم به اینصورت که پس از تایپ نام این متغیرهـا در خـط فرمـان Enter می‌زنیم. در ادامه برای تمرین بیشتر، توسط جدول (۱-۵) دستورات متلب متناظر بـا تعـدادی از عملیات ریاضی روی ماتریس‌ها را معرفی می‌کنیم.

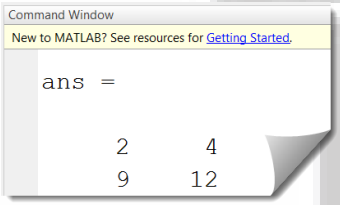
جدول ۱-۵- خلاصه دستورات

توضیحات	دستور متلب	عملیات ریاضی
جمع دو ماتریس	$A+B$	$A+B$
تفریق دو ماتریس	$A-B$	$A-B$
ضرب دو ماتریس	$A*B$	$A\times B$
محاسبه دترمینان	$det(A)$	$\|A\|$
به توان m رساندن یک ماتریس	$A\text{^}m$	A^m
c یک عدد ثابت است	$c*A$	$c\times A$
ترانهاده	$A'\ or\ transpose(A)$	A^T
معکوس ماتریس	$A\text{^}-1$ or $inv(A)$	A^{-1}
مقادیر ویژه λ مربوط به ماتریس A و I ماتریس همانی	$eig(A)$	$\|\lambda I-A\|=0$
دستور ترکیبی	$inv(A+B')$	$(A+B^T)^{-1}$
حذف سطرهای a تا b از ماتریس A	$A(a:b,:)=[\,]$	
حذف ستون‌های a تا b از ماتریس A	$A(:,a:b)=[\,]$	
معکوس تک تک مولفه‌های ماتریس A	$A.\text{^}-1$	
ضرب نظیر به نظیر مولفه‌های دو ماتریس A و B	$A.*B$	
ماتریس قطری با مولفه‌های روی قطر اصلی a و b	$C=[a\ b];$ $diag(C)$	$\begin{bmatrix}a & 0\\0 & b\end{bmatrix}$

توجه کنید ظاهر شدن علامت dot یا نقطه پیش از علائمی نظیر توان یا ضرب و تقسیم به معنای عملیات مولفه‌ای نظیر به نظیر است. به عنوان مثال با اجرای برنامه جدول (۱-۶)، نتایج بصورت شکل (۱-۸) در محیط پنجره‌ی فرمان متلب نمایش داده می‌شود.

جدول ۱-۶ –برنامه‌ی ضرب نقطه‌ای در متلب

```
clc;
clear all;
close all;
A=[1 2;3 4];
B=[2 2;3 3];
A.*B
```



شکل ۱-۸

که همانطور که مشاهده می‌کنیم نتیجه با ضرب معمولی ماتریسی متفاوت است و عملیات ضرب مولفه‌ای نظیر به نظیر صورت گرفته است. یعنی عنصر سطر اول و ستون اول از ماتریس A در عنصر سطر اول و ستون اول از ماتریس B ضرب شده ، یعنی عدد ۱ در عدد ۲ ضرب شده و مولفه‌ی سطر اول و ستون اول از ماتریس حاصل، ۲ شده است و ..... حاصل ضرب معمولی این دو ماتریس $\begin{bmatrix} 8 & 8 \\ 18 & 18 \end{bmatrix}$ و حاصلضرب ضرب نقطه‌ای این دو ماتریس $\begin{bmatrix} 2 & 4 \\ 9 & 12 \end{bmatrix}$ است. همانطور که مشاهده می‌کنیم این دو با هم متفاوت می‌باشند.

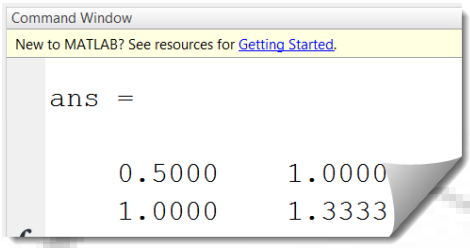
عملیات تقسیم در ماتریس‌ها تعریف نشده است. ولی چنانچه پیش از علامت تقسیم '/' از Dot یا نقطه استفاده کنیم، می‌توانیم عملیات تقسیم مولفه‌ای نظیر به نظیر را داشته باشیم به عنوان مثال به برنامه‌ی موجود در جدول (۱-۷) توجه کنید که این عملیات بر روی دو ماتریس A و B انجام گرفته است.

جدول ۱-۷ –برنامه‌ی تقسیم نقطه‌ای

```
clc;
clear all;
close all;
A=[1 2;3 4];
B=[2 2;3 3];
A./B
```

با اجرای برنامه‌ی جدول (۱-۷)، نتیجه بصورت شکل (۱-۹) در محیط پنجره‌ی فرمان نمایش داده می‌شود.

تلفن: ۲-۶۶۴۱۸۴۱۹۱



شکل ۱-۹

عنصر سطر اول و ستون اول از ماتریس A بر عنصر سطر اول و ستون اول از ماتریس B تقسیم شده ، یعنی عدد ۱ بر عدد ۲ تقسیم شده و مولفه سطر اول و ستون اول از ماتریس حاصل، 0.5 شده است و .....

محاسبه‌ی مقادیر ویژه‌ی ماتریس

مقادیر ویژه یک ماتریس با دستور eig محاسبه می‌شود. برای انجام این محاسبه کافیست پس از تعریف ماتریس مورد نظر، نام آن را در آرگومان دستور eig بیاوریم. برای آشنایی بیشتر با این دستور به مثال (۱-۳) توجه کنید.

مثال ۱-۳: ماتریس $A=\begin{bmatrix}1 & 2 & 3\\0 & 2 & 0\\3 & 2 & 1\end{bmatrix}$ موجود است. برنامه‌ای بنویسید که مقادیر ویژه ماتریس B بصورت $B=A^2+2A^{-1}+\begin{bmatrix}1 & 0 & 0\\0 & 2 & 0\\0 & 0 & 3\end{bmatrix}$ را محاسبه و در محیط Comand Window نمایش دهد.

حل: برنامه‌ی موجود در جدول (۱-۸) را در یک فایل Script یا M-فایل تایپ نموده و سپس آن‌را اجرا نمایید.

جدول ۱-۸ – برنامه‌ی مربوط به مثال (۱-۳)

```
clc;
clear all;
close all;
A=[1 2 3;0 2 0;3 2 1];
B=A^2+2*inv(A)+diag([1 2 3])
eig(B)
```

آشنایی با دستور rand

از این دستور برای تولید ماتریس با درایه‌های تصادفی استفاده می‌شود. با استفاده از دستور:

```
rand(a,b)
```

یک ماتریس با ابعاد $a \times b$ با توزیع تصادفی درایه‌ها در بازه‌ی $(0,1)$ ایجاد می‌گردد. اما در اکثر مواقع لازم است که داده‌های تصادفی را در یک بازه‌ی مشخص مثلا (a,b) تولید کنیم. به مثال (۱-۴) برای آشنایی با این گونه کاربردها توجه کنید.

مثال ۱-۴: برنامه‌ای بنویسید که ماتریس A را با ابعاد 3×5 با توزیع تصادفی هر یک از درایه‌ها در بازه‌ی $(-2,0)$ ، همینطور ماتریس B را با ابعاد 3×2 با توزیع تصادفی هر یک از درایه‌ها در بازه $(2,4)$ ، ایجاد نماید.

همانطور که در توضیح دستور rand گفته شد، این دستور درایه‌ها را با توزیع تصادفی در بازه‌ی $(0,1)$ تولید می‌نماید، حال آنکه در این مثال بازه‌های $(-2,0)$ و $(2,4)$ مد نظر است. برای حل این مساله نیاز به استفاده از چندجمله‌ای‌هایی از این دستور داریم به برنامه‌ی جدول (۱-۹) برای آشنایی با این عملیات توجه کنید.

جدول ۱-۹ – توضیح ماتریس با توزیع تصادفی درایه‌ها در بازه‌ی مورد نظر

```
clc;
clear all;
close all;

A=-2*rand(3,5)
B=2+2*rand(3,2)
```

با ترکیب ریاضی دستور $rand$ می‌توان بازه‌ی دلخواه و مورد نظر را تولید کرد. در برنامه‌ی جدول (۱-۹) توسط دستور $A = -2 * rand(3,5)$ یک ماتریس با ابعاد 3×5 تولید می‌شود که هر یک از مولفه‌های آن در بازه‌ی $(-2,0)$ قرار دارند و از طرفی تصادفی‌اند. با اجرای هر بار دستور rand یک ماتریس متفاوت تولید می‌شود. این دستور در مهندسی در مبحث شناسایی سیستم‌ها برای تحریک سیستم‌ها و یا هر جا که نیاز به یک ماتریس با توزیع تصادفی باشد، کاربرد دارد.

آرایه‌های سلولی

با ماتریس‌ها و یا آرایه‌های عددی و چگونگی تعریف آن‌ها آشنا شدیم، در این قسمت قصد داریم که با آرایه‌های سلولی آشنا شویم. به یاد داریم که در ماتریس‌های عددی، عناصر از اعداد تشکیل می‌شدند اما هر سلول در یک آرایه سلولی می‌تواند در حالت کلی شامل کاراکتر و عدد و یا ماتریس با ابعاد دلخواه باشد. در ادامه با آرایه‌های سلولی بیشتر آشنا می‌شویم.

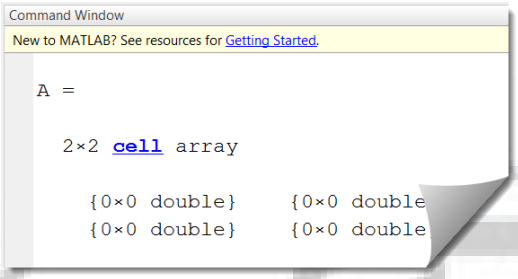
برای تعریف یک آرایه‌ی سلولی انتخاب‌های متنوعی داریم می‌توانیم از دستور cell استفاده کنیم یا از آکولاد '{}' استفاده کنیم. به مثال (۱-۵) برای آشنایی با کارکرد صحیح این دستور توجه کنید.

مثال ۱-۵ برنامه‌ای بنویسید که یک آرایه‌ی سلولی با ابعاد 2×2 را تعریف کند و سلول‌های سطر اول این آرایه را به ترتیب با عناصر کارکتری $motor1$ و $motor2$ و سلول‌های سطر دوم را به ترتیب با ماتریس‌های $\begin{bmatrix}1\\2\\3\end{bmatrix}$ و $\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\end{bmatrix}$ پر نماید.

برای تعریف یک آرایه‌ی سلولی با ابعاد خواسته شده می‌توانیم از دستور cell استفاده کنیم.

```
A=cell(2,2)
```

با اجرای این دستور نتیجه به صورت شکل (۱-۱۰) در محیط Command Window نمایش داده می‌شود.



شکل ۱-۱۰

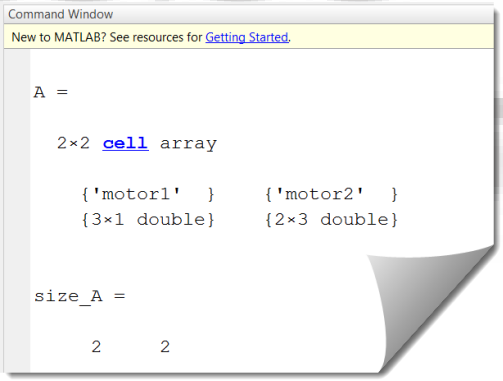
توسط برنامه‌ی جدول (۱-۱۰)، ابتدا یک آرایه‌ی سلولی خالی با ابعاد **2 × 2** تعریف شده و سپس سلول‌های سطر اول و دوم آن، با اطلاعات داده شده پر می‌شود و در نهایت توسط دستور size ابعاد این آرایه‌ی سلولی نمایش داده می‌شود.

- **روش اول**

جدول ۱-۱۰ – برنامه‌ی مربوط به آرایه‌ی سلولی در مثال (۱-۵) به روش اول

```
clc;
clear all;
close all;
A=cell(2,2);
A(1,1)={'motor1'};
A(1,2)={'motor2'};
A(2,1)={[1;2;3]};
A(2,2)={[1 2 3;4 5 6]}
size_A=size(A)
```

با اجرای برنامه جدول (۱-۱۰)، نتیجه بصورت شکل (۱-۱۱) در محیط پنجره‌ی فرمان نمایش داده می‌شود.



شکل ۱-۱۱