الأكاديمية العربية الدولية
Arab International Academy

إسم المادة : أساسيات الماتلاب

إسم المدرب : عبدالقادر عبدالله

الأكاديمية العربية الدولية – منصة أعد

الأكاديمية العربية الدولية
Arab International Academy

محتويات المحاضرة

- مقدمة عن MATLAB
- التعرف على واجهة MATLAB
- الأوامر الأساسية في MATLAB
- التعامل مع المتغيرات
- العمليات الحسابية الأساسية
- إنشاء المصفوفات
- التعامل مع المصفوفات
- الرسومات البيانية (Plotting)
- التعامل مع الدوال
- الحلقات الشرطية (if, else)
- الحلقات التكرارية (for, while)
- الملفات النصية في MATLAB
- الرسوم البيانية ثلاثية الأبعاد
- التعامل مع الصور
- إدارة المشاريع في MATLAB
- البرمجة الكائنية (Object-Oriented Programming)
- التعامل مع البيانات الكبيرة
- التطبيقات التفاعلية (Apps)
- تحسين الأداء (Performance Optimization)
- استخدامات MATLAB المتقدمة

مقدمة عن MATLAB

MATLAB هو برنامج متخصص في البرمجة الرياضية والحسابات التقنية.
يُستخدم على نطاق واسع في مجالات الهندسة والعلوم لتحليل البيانات والنمذجة.

يوفر أدوات قوية لحل المشكلات المعقدة باستخدام المعادلات الرياضية.
يتميز MATLAB بسهولة الاستخدام مقارنةً بلغات برمجة أخرى مثل C++ وPython .

يحتوي MATLAB على واجهة رسومية تسهل العمل مع البيانات والرسومات.

التعرف على واجهة MATLAB

تتكون واجهة MATLAB من عدة نوافذ مثل:

Command Window : لتنفيذ الأوامر الفورية.

Workspace: لعرض المتغيرات المستخدمة.

Current Folder: لإدارة الملفات والمجلدات.

Command History: لاستعراض الأوامر السابقة.هذه العناصر تجعل التفاعل مع البرنامج سلسًا وفعالًا.

الأوامر الأساسية في MATLAB

MATLAB يعتمد على الأوامر لتنفيذ المهام.

يمكنك بدء العمل بأوامر بسيطة مثل help للحصول على معلومات حول وظيفة معينة.

أوامر مثل clc لتنظيف شاشة الأوامر وclear لحذف المتغيرات مفيدة جدًا.

الأمر who يعرض جميع المتغيرات المتاحة في مساحة العمل.

يمكنك استخدام الأمر exit للخروج من البرنامج.

التعامل مع المتغيرات

المتغيرات في MATLAB هي الطريقة الأساسية لتخزين البيانات.

يتم إنشاء المتغيرات مباشرة عند تعيين قيمة لها، مثل: x = 5 .

يمكن تخزين أنواع مختلفة من البيانات، مثل الأعداد الحقيقية والمصفوفات.

لعرض قيمة المتغير، يمكن ببساطة كتابة اسمه في نافذة الأوامر.

الأمر clear يُستخدم لحذف المتغيرات وإفراغ مساحة العمل.

العمليات الحسابية الأساسية

MATLAB يوفر بيئة قوية للعمليات الحسابية.

يمكن تنفيذ العمليات الحسابية مثل الجمع (+)، الطرح (-)، الضرب (*)، والقسمة (/).

يمكن أيضًا استخدام أوامر مثل sqrt لحساب الجذر التربيعي وexp للأسس.

يتم تنفيذ العمليات بسهولة على الأعداد والمصفوفات بنفس الطريقة.

من الممكن استخدام الأقواس للتحكم في ترتيب العمليات الحسابية.

إنشاء المصفوفات

المصفوفات هي العنصر الأساسي في MATLAB .

يتم إنشاء مصفوفة بكتابة العناصر بين الأقواس المربعة مثل: A = [1 2; 3 4] .

يمكن للمصفوفات أن تكون أحادية أو متعددة الأبعاد، وتعامل بسهولة مع العمليات الحسابية.

للوصول إلى عناصر معينة، نستخدم الفهارس: A(1,2) للوصول إلى العنصر في الصف الأول والعمود الثاني.

المصفوفات تُستخدم في معظم تطبيقات MATLAB بسبب مرونتها وقوتها.

التعامل مع المصفوفات

MATLAB يُتيح تنفيذ العمليات على المصفوفات مثل الجمع والطرح وضرب المصفوفات.

العمليات العنصرية (.*, ./) تُستخدم لتنفيذ العمليات على كل عنصر بشكل مستقل.

الأمر transpose أو الرمز ' يُستخدم للحصول على مصفوفة منقولة.

يمكن تعديل أبعاد المصفوفات باستخدام الأمر reshape .

الأمر size يُظهر أبعاد المصفوفة (عدد الصفوف والأعمدة).

الرسومات البيانية (Plotting)

MATLAB يوفر أدوات قوية لإنشاء الرسومات البيانية بسهولة.

الأمر plot يُستخدم لرسم المنحنيات الثنائية الأبعاد.

يمكنك تخصيص الرسوم البيانية باستخدام ألوان وأنماط مختلفة مثل: plot(x, y, 'r--') .

الأمر title يُستخدم لإضافة عنوان، وxlabel وylabel لإضافة تسميات المحاور.

الرسوم البيانية تساعد في تصور البيانات وتحليلها بشكل فعال.

التعامل مع الدوال

الدوال في MATLAB تُستخدم لتجميع أكواد قابلة لإعادة الاستخدام.

يتم تعريف الدوال باستخدام الكلمة المفتاحية function تليها اسم الدالة والمعاملات.

يمكن تمرير قيم إلى الدالة وإرجاع قيم منها بسهولة.

على سبيل المثال، دالة لجمع رقمين:

```
function result = addNumbers(a, b)

result = a + b;

end
```

الحلقات الشرطية (if, else)

الحلقات الشرطية تُستخدم لتنفيذ الكود بناءً على شروط معينة.

إذا كانت الشروط صحيحة، يتم تنفيذ الكود داخل كتلة if .

يمكن استخدام elseif و else لتوفير خيارات إضافية.

الحلقات التكرارية (for, while)

تُستخدم الحلقات التكرارية لتنفيذ كتلة من الكود عدة مرات.

حلقات for تُكرر عددًا محددًا من المرات:

```
for i = 1:10
  disp(i);
end
```

بينما تستخدم while عندما يكون الشرط غير محدد :

```
while x < 10
  x = x + 1;
end
```

الملفات النصية في MATLAB

يمكنك قراءة وكتابة الملفات النصية بسهولة باستخدام MATLAB

الأمر fopen يُستخدم لفتح الملفات، وfclose لإغلاقها بعد الانتهاء.

الأمر fprintf يُستخدم لكتابة البيانات إلى ملف نصي، بينما fscanf وfgets لقراءة البيانات.

مثال لفتح وكتابة ملف نصي:

```
fileID = fopen('output.txt', 'w');
fprintf(fileID, 'Hello, MATLAB!');
fclose(fileID);
```

الرسوم البيانية ثلاثية الأبعاد

MATLAB يوفر دعمًا قويًا لإنشاء الرسوم البيانية ثلاثية الأبعاد.

الأمر mesh يُستخدم لرسم الشبكات ثلاثية الأبعاد، بينما surf يُظهر الأسطح.

يمكنك تخصيص الرسوم باستخدام الألوان والتدرجات لتمثيل البيانات بشكل أكثر وضوحًا.

الأوامر xlabel, ylabel, وzlabel تُستخدم لتسمية المحاور الثلاثة.

الرسوم ثلاثية الأبعاد تُستخدم لتحليل البيانات الأكثر تعقيدًا وتصورها.

التعامل مع الصور

يمكنك استخدام MATLAB لمعالجة الصور الرقمية بسهولة.

الأمر imread يُستخدم لقراءة صورة، وimshow لعرضها.

يمكن تطبيق فلاتر وتحسينات على الصور باستخدام دوال مدمجة مثل imfilter

.أيضًا يمكن تحويل الصور بين ألوان متعددة وتحليل تفاصيلها باستخدام MATLAB .

معالجة الصور مفيدة في التطبيقات الهندسية والطبية والبحثية.

إدارة المشاريع في MATLAB

يتيح MATLAB إدارة المشاريع بشكل منظم من خلال تنظيم الأكواد والملفات في مجلدات.

يمكنك استخدام المجلدات والمشاريع لتنسيق عملك وتجميع الأكواد المتعلقة بمشروع معين.

توفر MATLAB أدوات للتحكم في الإصدارات، مما يسمح بتتبع التغييرات في الأكواد.

إدارة المشاريع تضمن سهولة الوصول إلى الملفات والبيانات عند العمل على مشاريع كبيرة.

الأدوات المدمجة تجعل إدارة المشاريع بسيطة وفعالة.

البرمجة الكائنية (OOP)

يدعم MATLAB البرمجة الكائنية، مما يسمح بإنشاء كائنات وطرق لتحسين التنظيم.

يتم تعريف الكائنات باستخدام classdef ، ويمكن تضمين خصائص ودوال داخل الفئات.

البرمجة الكائنية تسهل إعادة استخدام الأكواد وتجنب التكرار.

مثال على إنشاء فئة:

```
classdef MyClass
properties
x
end
methods
function obj = MyClass(val)
obj.x = val;
end
end
end
```

التعامل مع البيانات الكبيرة

MATLAB يوفر أدوات لتحليل البيانات الكبيرة مثل Tall Arrays للتعامل مع المصفوفات الكبيرة جدًا.

يمكنك استخدام الأوامر مثل tall لتحويل البيانات الكبيرة إلى نوع خاص من المصفوفات القابلة للمعالجة على مراحل.

يُستخدم datastore لقراءة وتحليل مجموعات البيانات الضخمة مثل النصوص والصور بشكل مجزأ.

هذه الأدوات تساعد في تحليل البيانات الكبيرة دون الحاجة إلى تحميلها بالكامل في الذاكرة.

MATLAB يتكيف مع الموارد المتاحة لضمان كفاءة معالجة البيانات الكبيرة.

التطبيقات التفاعلية (Apps)

MATLAB يوفر أدوات لإنشاء تطبيقات تفاعلية باستخدام App Designer.
يمكنك بناء واجهات مستخدم رسومية GUI بسهولة عبر السحب والإفلات للعناصر المرئية.

يتيح App Designer إضافة الوظائف البرمجية إلى عناصر الواجهة للتفاعل مع المستخدم.
يمكن استخدام التطبيقات لعرض البيانات، تعديل المدخلات، وتوفير أدوات تحليل مخصصة.

إنشاء التطبيقات يتيح للمستخدمين النهائيين التفاعل مع MATLAB بدون الحاجة لمعرفة البرمجة.

تحسين الأداء (Performance Optimization)

MATLAB يوفر عدة أدوات لتحسين أداء الأكواد مثل profiler لتحليل الكود واكتشاف الأجزاء البطيئة.

يمكن تحسين الأكواد باستخدام المتجهات بدلاً من الحلقات، مما يقلل من زمن التنفيذ.

الأمر parallel يسمح بتوزيع العمليات على عدة معالجات لزيادة السرعة.ت

ُستخدم الدوال المدمجة مثل parfor وspmd للتوازي وتحسين الأداء في البرامج الكبيرة.

تحسين الأداء مهم عند التعامل مع التطبيقات الحسابية الكبيرة والمعقدة.

استخدامات MATLAB المتقدمة

MATLAB هو أداة قوية وشاملة للتحليل والنمذجة والبرمجة العلمية والهندسية.
يغطي MATLAB مجموعة واسعة من التطبيقات في مجالات مثل معالجة الصور، الذكاء الاصطناعي، والأنظمة الديناميكية.

بفضل مكتباته الغنية وسهولة الاستخدام، يعد MATLAB خيارًا ممتازًا للمبتدئين والمحترفين.
باستخدام MATLAB ، يمكن للمستخدمين الانتقال من النماذج البسيطة إلى المشاريع المعقدة بسهولة وكفاءة.

تعلم الأساسيات هو الخطوة الأولى نحو استكشاف القدرات المتقدمة في MATLAB.