



كلية العلوم

القسم : الرياضيات

السنة : الرابعة

المادة : برمجة غرضية التوجه

المحاضرة : الثالثة / نظري /

{{ مكتبة A to Z }}

مكتبة A to Z Facebook Group :

كلية العلوم ،

يمكنكم طلب المحاضرات برسالة نصية (SMS) أو عبر (What's app-Telegram) على الرقم 0931497960



جامعة طرطوس
كلية العلوم
قسم الرياضيات
مقرر: برمجة غرضية التوجه- السنة الرابعة
المحاضرة الثالثة

الرسوم البيانية

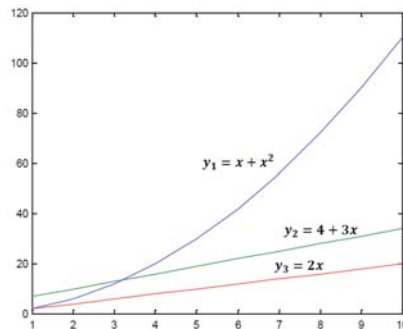
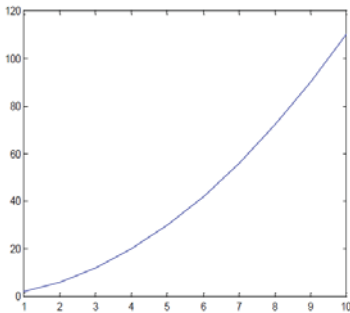
- ❖ يملك ماتلاب إمكانيات هائلة لعرض المتجهات والمصفوفات كرسوم، وتذليلها بالتعريفات كعنوان الرسم، وتسمية المحاور والمنحنيات،
- ❖ وبعد ذلك طباعة هذه الرسوم طباعة بجودة عالية.

أمر الرسم

- ❖ تقوم دالة **plot** بعدة مهام اعتمادا على مدخلاتها فلو كان **y** متجها فالأمر **plot(y)** يولد رسما للمتجه يفترض فيه أن **x** هي الرمز الدليلي لكل عنصر في المتجه. أما الأمر **plot(x,y)** فيرسم العلاقة بين هذين المتغيرين في مستوى **(x,y)** حيث تمثل **y** قيم المحور الرأسي و **x** قيم المحور الأفقي فمثلا ، نستخدم الأوامر التالية لرسم هذه الدالة التربيعية: $y = x + x^2$

```
x = 1:10;  
y = x + x.^2;  
plot(x,y)
```

- ❖ لاحظ أننا استخدمنا النقطة (.) قبل معامل الرفع للأس x^2 ؛ لأننا نريد رفع كل عنصر في المتجه **x** إلى الأس ٢



- ❖ ويمكن رسم أكثر من علاقة في نفس الوقت بتكرير المدخلات المطلوب رسمها كما هو موضح في المثال التالي

```
y2 = 4 + 3*x;  
y3 = 2*x;  
plot(x,y,x,y2,x,y3)
```

أمر الرسم

- ❖ وإذا كان المطلوب رسم أكثر من منحنى، فيقوم ماتلاب بشكل تلقائي بتلوين المنحنيات بألوان مختلفة. ومن الممكن أن يقوم المستخدم بتحديد لون مميز و (أو) علامات مميزة (+، ×) أو خط معين (متصل أو متقطع) لكل منحنى.
- ❖ والمثال التالي يوضح كيف يتم تلوين المنحنى الأول باللون الأصفر وخط مقطع وعلامة +:

`plot(x, y, 'y:+')`

- ❖ لاحظ أن من الألوان المقبولة `c m y r g b w k` وتمثل الألوان الأسود والأبيض والأزرق والأخضر والأحمر والأصفر والأرجواني والأزرق الفاتح على التوالي. ومن الخطوط المقبولة - (متصل)، - - (مقطع)، : (منقط)، - . (مقطع منقط)، `none` (عدم وجود خط). ومن العلامات المقبولة + ، o ، × ، *

<<< تنبيه >>>

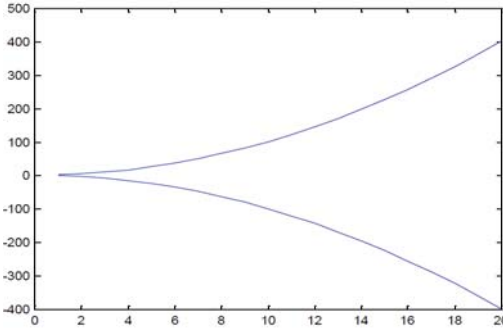
- ❖ بالإضافة إلى أمر الرسم `plot` يوجد أوامر أخرى لرسم أنواع مختلفة من الرسوم مثل الأوامر `bar` و `area` و `hist` و `pie` وغيرها.

نافذة الرسم

- ❖ يقوم ماتلاب برسم الرسوم البيانية في نافذة الرسم. وتقوم دوال الرسم بفتح نافذة الرسم وإرسال الرسم إليها، أو إرسال الرسم إلى نافذة الرسم المفتوحة حالياً. وإذا كان هناك نافذة رسم موجودة، وأردت أن يكون الرسم الثاني في نافذة أخرى استخدم الأمر: `figure`
- ❖ فهذا الأمر يقوم بفتح نافذة رسم جديدة. أما الأمر: `figure(n)`
- ❖ فيقوم بإرسال الرسم إلى النافذة المفتوحة التي تحمل الرقم `n`

إضافة رسم على الرسم الحالي

- ❖ يتيح الأمر `hold on` إضافة رسم منحنى أو منحنيات إلى نافذة رسم واحدة. فعندما تكتب: `hold on`
- ❖ يقوم ماتلاب بإضافة الرسم التالي لهذا الأمر فوق الرسم الموجود أصلاً في نافذة الرسم، مع تعديل المحاور إن تطلب الأمر ذلك (بدون هذا الأمر، يقوم ماتلاب بإزالة الرسم السابق ووضع الرسم الجديد مكانه). وهذا المثال يوضح عمل هذا الأمر:



```
x = 1:20;  
y1 = 2 - x.^2;  
plot(x, y1)  
hold on  
y2 = 2 + x.^2;  
plot(x, y2)
```

- ❖ تابع ماذا يحدث في نافذة الرسم حتى تحصل على هذا الرسم في الشكل الآتي

تكوين رسوم صغيرة داخل نافذة الرسم

- ❖ تمكن دالة `subplot` من عرض أكثر من رسم في نافذة رسم واحدة، وكذلك طبعتها في صفحة

`subplot(m,n,p)`

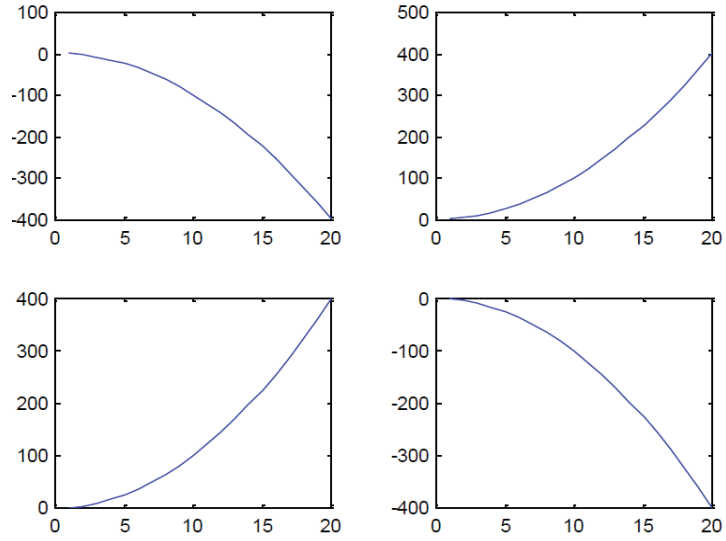
واحدة؛ فكتابة

- ❖ تقوم بتقسيم نافذة الرسم إلى مصفوفة (`m x n`) من الرسوم الصغيرة، مع إعطاء المكان `p` للرسم التالي؛ حيث يتم ترتيب الرسوم في النافذة من اليسار إلى اليمين، ثم من الأعلى للأسفل؛

تكوين رسوم صغيرة داخل نافذة الرسم

❖ فعلى سبيل المثال، تقوم الأوامر التالية برسم أربعة رسوم صغيرة في نافذة واحدة، موضحة في الشكل

```
x = 1:20;  
y1 = 2 - x.^2;  
y2 = 2 + x.^2;  
y3 = x.^2;  
y4 = -x.^2;  
subplot(2,2,1); plot(x,y1);  
subplot(2,2,2); plot(x,y2);  
subplot(2,2,3); plot(x,y3);  
subplot(2,2,4); plot(x,y4);
```



<<< تنبيه >>>

يمكن كتابة أكثر من أمر في سطر واحد بشرط أن يفصل بينهما بفاصلة عادية (،) أو منقوطة (؛)

التحكم بالمحاور

❖ للدالة **axis** العديد من الخيارات التي تسمح بالتحكم في مقياس واتجاه المحاور. وبدون تحديد أي خيارات يقوم ماتلاب باختيار قيمة المحاور مستخدماً القيم الكبرى والصغرى في البيانات المراد رسمها.

❖ لكن تحديد هذه الخيارات من قبل المستخدم يجبر ماتلاب على اعتمادها في الرسم. فالأمر: **axis([xmin xmax ymin ymax])**

❖ يلزم ماتلاب باستخدام القيم بين القوسين لبداية ونهاية محوري **x** و **y** أما الأمر: **axis square**

❖ فيجعل الفرق بين كل قيمتين في المحورين متساوية.

❖ والأمران التاليان يتحكمان في مسألة ظهور أو اختفاء المحاور من الرسم:

axis on
axis off

grid off
grid on

❖ أما الأمر:

❖ فيتحكمان في ظهور الخطوط الأفقية والرأسية في الرسم.

تسمية عنوان الرسم والمحاور

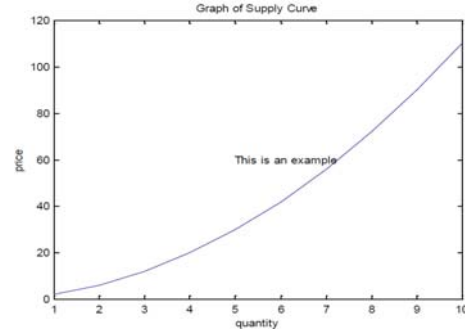
❖ تضيف الدوال الثلاث (**xlabel,ylabel,zlabel**) أسماء المحاور للرسم.

❖ أما الدالة **title** فتضيف عنوان الرسم في أعلاه.

❖ وتسمح الدالة **text** بكتابة نص في مكان ما في نافذة الرسم؛ والمثال التالي يوضح عمل هذه الدوال:

تسمية عنوان الرسم والمحاور

```
x = 1:10;  
y = x + x.^2;  
plot(x,y)  
xlabel('quantity')  
ylabel('price')  
title('Graph of Supply Curve')  
text(5,60,'This is an ex  
ample')
```



❖ لاحظ أن النصوص توضع دائما بين علامتي تنصيص، ولاحظ أيضا أننا طلبنا من

ماتلاب عند استخدام الأمر **text** أن يضع النص عند الإحداثي (٥,٦٠)

مفتاح الرسم البياني

❖ يضع الأمر **legend** صندوق صغير داخل الرسم، يبين فيه

مسمى كل منحنى يظهر في نافذة الرسم **legend('Supply')**

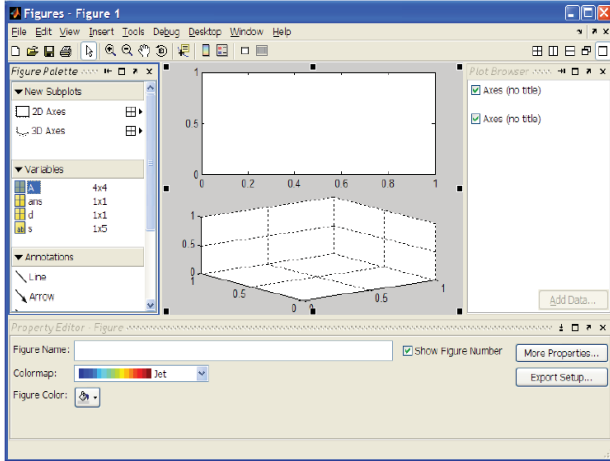
أدوات الرسم

❖ تتضمن بيئة ماتلاب أدوات رسم إضافية تسهل بشكل كبير من تحرير

رسومات ماتلاب حسب تفضيلات المستخدم، ويمكن الوصول لهذه

الأدوات من خلال نافذة الرسم؛ أو من خلال الأمر، **plottools**.

ويبين الشكل نافذة أدوات تحرير الرسومات



التحكم بالانسحاب والتكرار

❖ يمكن التحكم بالانسحاب بواسطة البلاغات الشرطية، والتحكم في تكرار تنفيذ أمر أو مجموعة من الأوامر بواسطة الحلقات التكرارية.

❖ ويشتمل ماتلاب على عدة طرق للتحكم في البلاغات الشرطية مع أو بدون الحلقات التكرارية، أهمها:

١ . بلاغ الشرط.

٢ . بلاغ الانتقال.

٣ . التكرار المحدود.

٤ . التكرار غير المحدود.

٥ . بلاغ وقف التكرار.

بلاغ الشرط

❖ تقوم الجملة الشرطية (**if**) بتقييم التعبيرات المنطقية؛ ومن ثم تنفيذ أمر أو مجموعة من الأوامر إذا كانت التعبيرات المنطقية صحيحة.

❖ وتستخدم كلمة **end** لتحديد نهاية عمل الجملة الشرطية؛ ومن أبسط صيغها:

```
A = 2;
```

```
if A == 2
```

```
disp('A تساوي 2')
```

```
end
```

❖ في هذا المثال يتحقق ماتلاب من صحة التعبير المنطقي، فإن كان صحيحا (وهو صحيح في هذا

المثال) عرض الجملة (A تساوي 2) على الشاشة باستخدام الأمر **disp**. ولو لم تكن A=2، فلن

يعرض شيئا .

❖ ويبين هذا المثال أن ماتلاب يمكن أن يتعامل مع نصوص اللغة العربية

بلاغ الشرط

❖ ويمكن استخدام الشرطين **elseif** و **else** لتنفيذ أوامر أخرى في حالة عدم صحة شرط التعبير المنطقي الخاص بالشرط **if** مثال:

```
A = 3;
if A == 2
disp('A تساوي 2')
elseif A == 0
disp('A تساوي صفر')
else
disp('A لا تساوي 2')
end
```

❖ في هذا المثال يتحقق ماتلاب من صحة تعبيرين منطقيين،
❖ فإن كان أحدها صحيحا عرض الجملة المناسبة على الشاشة.
❖ أما إذا كانا غير صحيحين فينفذ الأمر الذي يلي **else**، وهو عرض رسالة (A لا تساوي 2).
❖ ومن المهم أن تدرك أن هذه الشروط ومعاملات العلاقات المنطقية تعمل مع الأعداد المفردة بطريقة مختلفة عن المصفوفات.
❖ فللتأكد من تحقق المساواة بين متغيرين مفردين يمكن أن تستخدم: **if a == b, ...**

❖ أما في حالة المصفوفات فالشرط يتأكد من تساوي كل عنصر مع ما يقابله، وليس تساوي المصفوفات ككل (بشرط تساوي حجم المصفوفات). والطريقة السليمة للتحقق من تساوي المصفوفات هي استخدام دالة **isequal** على هذا النحو: **if isequal(A,B), ...**

بلاغ الانتقال بين الحالات

❖ تنفذ العبارة **switch** مجموعة من العبارات اعتمادا على قيمة المتغير أو التعبير.
❖ ودور الكلمتين **case** و **otherwise** هو تقسيم هذه المجموعات إلى حالات مختلفة.
❖ وأول حالة تتطابق مع الشرط يتم تنفيذها فقط، ويجب أن يكون هناك كلمة **end** كالمعتاد لتحديد متى يتوقف ماتلاب عن التحقق من الشرط.

بلاغ الانتقال بين الحالات

```
A = 3;
switch A
case 2
```

❖ ويمكن التعبير عن مثال بلاغ الشرط السابق باستخدام هذا الشرط:

```
disp('A تساوي 2')
case 0
disp('A تساوي صفر')
otherwise
```

❖ **for** يقوم بتكرار تنفيذ مجموعة من التعبيرات لعدد محدد من المرات. ويجب أن يقابلها ما يفيد انتهاء التعبيرات الداخلة في حلقة التكرار وهو كالمعتاد كلمة مثال:

```
disp('A لا تساوي 2')
end
n = 3;
for i = 1:10
r(i) = i*n;
end

r =
3 6 9 12 15 18 21 24 27
30
```

❖ فهذه الحلقة التكرارية تقوم بتكوين متجه صف مكون من عشرة عناصر.
❖ وفي كل تكرار تقوم الحلقة بتكوين عنصر جديد في هذا الصف يساوي الرمز الدليلي **i** مضروبا في العدد **n** الذي يساوي 3.
❖ ويتم حفظ النتيجة في المتجه **r**

التكرار لعدد محدود من الدورات

❖ وتعد طريقة كتابة التكرار مع ترك مسافات مناسبة عادة حسنة عند البرمجة لتسهيل القراءة وللتمييز بين مختلف الحلقات التكرارية.

```
m = 5; n = 3;
```

```
for i = 1:m
```

```
    for j = 1:n
```

```
        H(i,j) = 1/(i+j);
```

```
    end
```

```
end
```

❖ يقوم محرر ماتلاب بهذه المهمة بشكل تلقائي.

❖ توضح الحلقات التكرارية المتداخلة في المثال التالي قائمة ترك المسافات:

❖ تقوم هاتان الحلقتان التكراريتان المتداخلتان بتكوين المصفوفة **H** ذات الأبعاد (3 × 5)، وكل

عنصر فيها يساوي الواحد صحيح مقسوما على جمع الرمزتين الدليليين الخاص بذلك العنصر.

التكرار لعدد غير محدود من الدورات

❖ تقوم **while** بتكرار تنفيذ تعبير أو مجموعة من التعبيرات عددا لا نهائي من المرات حتى يتحقق الشرط المنطقي المطلوب تحققه.

❖ وكالعادة يجب أن يوجد كلمة **end** لتحديد الجزء من البرنامج الداخل في عملية التكرار؛ والمثال التالي يوضح عمل هذه الحلقة التكرارية:

```
a = 0.1;
```

```
while a^2 < 100
```

```
    a = a + 1;
```

```
end
```

```
a
```

```
a =
```

```
10.1000
```

❖ فالمطلوب هنا أن يفترض ماتلاب ابتداء أن قيمة المتغير **a** هي 0.1 ،

❖ ثم يبدأ بإضافة العدد 1 إليه ويجمع الناتج مع 1 مرة أخرى،

❖ ويستمر في هذه العملية مادام أن مضروب **a** في نفسها أقل من 100 . وستلاحظ أن التكرار

سوف يتوقف عندما تصل قيمة **a** إلى 10.1 حيث إن مضروب هذا العدد في نفسه يتجاوز 100

بلاغ وقف التكرار

❖ تستخدم كلمة **break** مع **for** و **while** لوقف عملية التكرار،

❖ وتعد عملية وقف التكرار مهمة للخروج من التكرار عندما لا يكون هناك فائدة من استمراره (خاصة

في حالة التكرار غير المحدود)؛

❖ والمثال التوضيحي يبين ذلك:

```
a = 0; b = 3;
```

```
while b-a > 0.01*b
```

```
    x = (a+b)/2;
```

```
    if x == 1.5,
```

```
        break
```

```
    end
```

```
end
```

❖ فنحن نعلم من قيم **a** و **b** أن قيمة المتغير **x** تساوي 1.5 وأن **b-a** لا يمكن أن تزيد عن **0.01*b**، ولهذا

سيستمر التكرار إلى ما لا نهاية إذا لم نقم بتوقيفه باستخدام بلاغ وقف التكرار.

معاملات العلاقات

❖ المعاملات التي تستخدم مع بلاغات الشرط لتحديد طبيعة العلاقة بين متغيرين في لغة ماتلاب هي:

المعنى	استخدامه	المعامل
A تساوي B	A==B	==
A أكبر من B	A>B	>
A أصغر من B	A<B	<
A أكبر من أو تساوي B	A>=B	>=
A أصغر من أو تساوي B	A<=B	<=
A لا تساوي B	A~=B	~=

المعاملات المنطقية

❖ المعاملات المنطقية التي تستخدم مع بلاغات الشرط في لغة ماتلاب هي:

المعنى	استخدامه	المعامل
B و A	A & B	&
B أو A	A B	
ليس A	~A	~