

```
>> poly(ans)
```

```
ans =
```

```
1 0000 3 0000 -15 0000 -2 00
```

ضرب وقسمة كثير الحدود:

لضرب معادلتين كثير حدود في بعضهما استعمال دالة conv وللقسمة الدالة deconv

```
>> x
```

```
x =
```

```
1 3 -15 -2 9
```

```
>> z
```

```
z =
```

```
1 0 0 0 -2
```

```
>> mu = conv(x, z)
```

```
mu =
```

```
1 3 -15 -2 7 -6 30
```

```
>> [d, r] = deconv(mu, x)
```

عند استعمال deconv لقسمة كثيري حدود فإنه ينتج متجهين:

- الأول d ناتج القسمة.
- الثاني r باقي القسمة) وفي المثال السابق كان الباقي من القسمة متجه صفري.

يمثل عدد الأعمدة

الدرس الخامس:

أوامر مفيدة في MatLab

وقفة قصيرة من الأمور الرياضية ودوالها التي تكلمنا عنها في الدروس السابقة، نتعلم المزيد عن كيفية استعمال matlab والأوامر الأساسية به .

مسح إطار الأوامر:

أثناء عملنا قد نرغب من وقت لآخر في مسح كل ما هو موجود على إطار الأوامر. يوجد طريقتين لذلك:

1. إذا كنت من محبي استعمال الفأرة أختار Edit-> Clear Command Window
2. أما إذا كنت تفضل استعمال لوحة المفاتيح فأكتب `clc` ثم Enter.

ملاحظة:

مسح إطار الأوامر لن ي حذف المتغيرات التي تم تعريفها خلال جلسة العمل، أنظر لإطار جلسة العمل *Workspace* ستجد أن المتغيرات لم تتغير أو تحذف. يمكنك أيضا استعمال الأمر `Whos` لعرض المتغير المعرفة في جلسة العمل الحالية للتأكد، إذا كان إطار جلسة العمل غير ظاهر لديك .

حذف جميع المتغيرات المعرفة في جلسة العمل:

وهنا أيضا لدينا طريقتين:

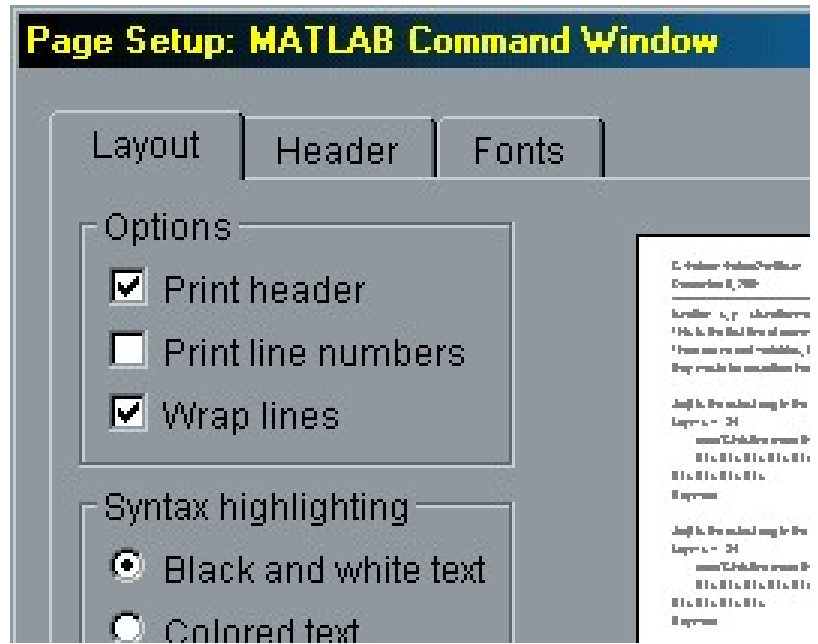
1. إذا كنت من محبي استعمال الفأرة أختار Edit-> Clear Workspace
2. أما إذا كنت تفضل استعمال لوحة المفاتيح فأكتب `clear` ثم Enter.

طباعة محتويات إطار الأوامر:

طباعة محتويات إطار الأوامر كاملة اختار File -> Print

ولطباعة الجزء المحدد فقط من الإطار اختار File -> Print selection

وللتحكم في تنسيق المخرجات من الطباعة اختار File -> Page Setup حيث تظهر لك مربع حوار page setup الذي يمكن من خلاله التحكم في تنسيق الصفحة مثل ظهور رأس الصفحة Header أو لا ومحتويات هذا الرأس، ظهور أرقام للأسطر، والخطوط fonts المستعملة أثناء الطباعة.



ملاحظات مفيدة:

- خلال عملك على matlab تذكر أنه حساس لحالة الأحرف case sensitive ، لذلك فإن Clear ليست مثل clear على سبيل المثال .
- يمكن أن تكتب أكثر من أمر على سطر واحد في MatLab شرط أن تفصل بينهما بفاصلة منقوطة.

```
>> A = [1 2 3 4 5]; B = [6 7 8 9 10];
>> C = A + B
C
7 9 11 13 15
>>
```

كما يمكن كتابة الأمر الواحد على سطرين منفصلين، خاصة إذا كان عرض الشاشة لا يتسع له) بأن نضع ثلاث نقاط (...) عند نهاية السطر الأول.

```

>> D = [ 2 5 2 4 1 66 8 44 88 66 ...
5 7 44 88 44 787 56 66 4]

D
Columns 1 through 12
2 5 2 4 1 66 8 44 88 66 5 7
Columns 13 through 19
44 88 44 787 56 66 4
>>

```

خلال العمل على MatLab فإن الأوامر التي تكتبها في إطار الأوامر تحفظ في حافظة الـ History وقد تسأل ما الفائدة من هذا؟

الفائدة منه أنه يمكنك إعادة استدعاء إي من هذه الأوامر السابقة وتنفيذها من جديد، وذلك من خلال الضغط المتكرر على مفتاح السهم للأعلى حتى تصل إلى الأمر الذي تريد تكراره، وذلك دون الحاجة إلى إعادة كتابته مرة ثانية.

- للحصول على المساعدة حول أي أمر أو دالة في MatLab مباشرة في إطار الأوامر أكتب `help` ثم اسم الأمر أو الدالة وسوف تظهر لك كل المعلومات التي تريدها حول ذلك الأمر، جرب مثلاً `help sin`

الدرس السادس:

البرمجة في MatLab

كما ذكرنا في الدرس الأول من هذه السلسلة فإن MatLab هو بيئة تطوير برمجية تحوي العديد من الدوال الجاهزة، بالإضافة إلى إمكانية كتابة برامج ودوال خاصة بنا حسب الحاجة. خلال هذا الدرس سوف نتعرف على الأوامر البرمجية في MatLab.

الجملة الشرطية: if

تستخدم للاختيار بين أمرين حسب شرط محدد

الصيغة العامة:

```
if <condition>
    <program1>
else
    <program2>
end
```

في حالة تحقق الشرط condition يتم تنفيذ الكود في program1 وإذا لم يتحقق الشرط يتم تنفيذ الكود في program2

مثال:

```
>> if n < 0
    disp('n is negative')
else
    disp('n is positive')
end
n is positive
>>
n
=
71
>>
```

يمكن أن تأخذ جملة if شكلا أكثر تداخلا باستعمال أكثر من مستوي لـ elseif

```
if expression1
    statements1
elseif expression2
```