

# ECE Linux Group

-

2002-

## مقدمة عن المجموعة

### ركائز وأهداف المجموعة

- أهم أهداف مجموعة مستخدمي "لينكس" هي مشاركة المعرفة والخبرة بين أعضاء المجموعة.
- ويندرج تحت هذا الهدف ما يلي:
  - العمل الجماعي المكون من شخصيات من مختلف الأعمار والثقافات.
  - تعلم حل المشكلات وهي احد ركائز التعليم( نحن في مجموع لينكس لا نكلف شركات تجاربه لحل مشاكلنا؟!).
  - تعلم جمع المعلومات المفيدة في تنصيب وتعريف البرامج.
  - تعلم صيانة الاجهزة (hard ware).
  - مشاركة بقية مجموعات لينكس المنتشرة حول أنحاء العالم.

### أنشطة المجموعة

- تأسست المجموعة في كلية اتصالات الهندسة في أماره الشارقة بدوله الإمارات العربية المتحدة في يوم الاثنين الموافق 5-أكتوبر-2002
- ووضعنا لها الأنشطة التالية:
  - دروس تعليمية لكل من نظامي " لينكس " و"يونكس"(Linux & Unix) في كل لقاء أسبوعي لرفع مستوى الخبرة لفريق المجموعة.
  - تنصيب تطبيقات وبرامج لينكس أضافه إلى الخوادم (servers).
  - تنصيب و تعريف الأنواع المختلفة من الخوادم(servers)مثل: (ssh,samba,NIS,database.mail.web,ftp) وغيرها.
  - تنصيب و تعريف الشبكات(network).
  - الأمن في الشبكات والخوادم.
  - تشغيل الخوادم, كما يمكن لأي فرد من المجموعة أن يشارك في أي نشاط يجده مناسباً لنعم الفائدة على الجميع.

### خطط مستقبلية قريبة

- تم اقتراح عدد من الخطط المستقبلية القريبة في اللقاء الأول ،منها:
- تأسيس مجتمع متكامل لهذه المجموعة بحيث يحدد مسؤوليات للمتطوعين من المجموعة للإعلان عن المجموعة وتنظيم المواعيد.
- تأسيس فريق عمل لنشر أنشطة المجموعة عبر الانترنت.
- تأسيس رابطته متكاملة لمجموعه لينكس.
- امكانيه دعوه خبراء ومتخصصين لعقد ندوات ومؤتمرات في الكلية.

### خطط مستقبلية غير بعيدة

- إنشاء مجله لينكس لتوثيق أنشطة المجموعة.
- فتح باب المشاركة الخارجية بعد سماح الكلية بذلك.
- تصدير أفكار المجموعة لجميع مؤسسات الدولة وجامعاتها .
- المشاركة في تعريب نظام لينكس.
- تنظيم لقاءات ومؤتمرات إقليميه ودوليه.

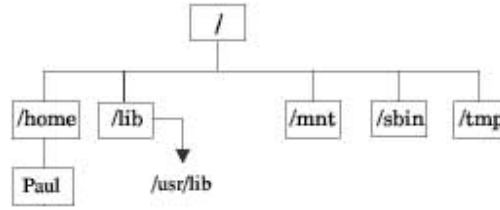
أحمد بنطية  
مؤسس مجموعة لينكس بكلية اتصالات للهندسة  
دولة الإمارات العربية المتحدة  
الشارقة

الدرس الأول: مقدمة عن سطر الأوامر وهيكله نظام "لينكس"

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

### معلومات عامة

- Command line : عبارة عن سطر للأوامر يتم معالجته من قبل النظام سطرًا بعد سطر (interpreter) كما هو الحال في Dos
- الصيغة العامة:
- \$ command <-option> <File name>
- أساس النظام يبدأ من "root" (/) ويندرج تحته الملفات والمجلدات



### بعض الأوامر المفيدة (Command line)

- 1 ls (List): يستخدم لعرض محتويات المكان الذي أنت فيه
  - أمثلة:
  - للعرض العادي
  - للعرض المفصل
  - للعرض المفصل بالإضافة إلى الملفات الخفية
  - لملفات الخفية: ملفات يسبق تسميتها نقطة (.). مثل: (.profile). يستخدم غالبًا لوضع معلومات عن مميزات النظام (configuration)
- 2 less more: يستخدم لعرض محتويات ملف مثل ملفات الكتابة (صفحة بعد صفحة) وهي بذلك تشابه الأمر
- 3 cat: استخدم لعرض محتويات ملف مثل ملفات الكتابة (دفعة واحدة)
- 4 df: يستخدم لعرض معلومات عن القرص الصلب (hard disk)
- 5 halt: يستخدم لإغلاق النظام

### معلومات متنوعة (من أسئلة الطلاب المشاركين)

- تقسيم القرص الصلب يقلل من مساحة الملف بمعنى: إذا كان القرص الصلب لا يحتوي إلا على قسم واحد (C:\) فإن أي ملف يشغل مساحة أكبر مما إذا كان مقسمًا إلى قسمين أو أكثر مثل (d, \:c) )
- في نظام "لينكس" الرمز @ الذي يأتي بعد اسم الملف يدل على أن اسم الملف ليس حقيقي وإنما هو عبارة عن رابطة (link) للملف الحقيقي (له نفس معنى shortcut في windows)
- وسائل الإدخال (input drivers) مثل القرص المرن واقرص الليزر (floppy, cdrom) تجدها تحت الملف /mnt (للعرض: ls /mnt)

- الأمر mount (خاص بمن لديه كلمة المرور ل: root) يمكن المستخدم من وضع او تنصيب وسائل الإدخال مثل: cdrom, floppy
  - مثال:
- mount /mnt/floppy
- وعكسه (ازالة التنصيب):
  - مثال:
- umount /mnt/floppy
- تدرج تحت /dev اقسام القرص الصلب مثل: /dev/hda1 حيث:
  - hda1: قرص صلب رقم 1 مكون من قسم واحد (hd: hard disk , a: the name of the hard disk , 1:the number of the partitions)
  - ومثله hdb2: قرص صلب رقم 2 مكون من قسمين (two partitions)
- الموديم الداخلي (internal modem) يعتبر موديم غير حقيقي، بمعنى ان معظمهم لا يعمل على نظام "لينكس"، بينما الموديم الخارجي (external modem) يعتبر موديم حقيقي يعمل على جميع أنظمة التشغيل

إعداد أعضاء مجموعة "لينكس" بإشراف الدكتور أحمد بنطية  
مؤسس مجموعته لينكس بكلية اتصالات للهندسة  
دولة الإمارات العربية المتحدة  
الشارقة

### الدرس الثاني: صيانة الأجهزة وتعلم PATH

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

#### معلومات عامة

- Basic input output system (BIOS) : هو النظام الرئيسي الذي تستطيع من خلاله تعريف أجزاء الحاسوب، لا يخزن في القرص الصلب وإنما يخزن في دائرة تعرف بـ: CMOS ، يفتح مباشرة عند تشغيل الجهاز بالضغط على F2 او DEL او غيره (يعتمد على نوع الحاسوب، وتستطيع معرفته من الكتب المرفقة بالحاسوب)
- يقوم BIOS عند التشغيل باختبار الأجزاء التالية على الترتيب:
  - المعالج (CPU)
  - بطاقة الفيديو
  - الذاكرة
  - لوحة المفاتيح
  - القرص المرن
  - القرص الصلب
  - ثم يشغل الجهاز
- برنامج Ghost : برنامج يستخدم لعمل صورة للقرص الصلب (image)، فعند اصابة الحاسوب بفيروس يمكنك استرجاع النظام بتشغيل البرنامج من "DOS"، له نفس وظيفة restorer في windows، ولكن عند اصابة windows بفيروس كيف ستستخدم restorer

#### صيانة الأجهزة

- عند تركيبك للقرص الصلب (إذا كنت من هواة تجميع الحاسوب) عليك مراعاة نوع التوصيل مع اللوحة الأم (mother board)، بمعنى ان تجعل قرصك الصلب هو القرص الرئيسي (master) إذا كان لديك أكثر من واحد، او تجعله ثانوي (secondary = slaver)، او تجعله منفرد إذا لم يكن لديك إلا واحد (single)، او تجعل النظام (operating system) يحدد نوع التوصيل تلقائياً (cable select)
- ستجد هذه المصطلحات الإنجليزية مكتوبة على القرص الصلب لتقوم انت بالاختيار

#### بعض الأوامر المفيدة (Command line)

- 6 alias : يستخدم لتغيير اسم الأمر (ترميز)
  - أمثلة:
 

```
alias dir="ls -l"
```
  - فعند الضغط الآن على dir يكون الأمر هو ls -l ، وتستطيع ان تضغط ايضا على ls -l ولا تحذف، مع إعادة تشغيل النظام يتم الغاء جميع الرموز التي استخدمتها، ولكي تحافظ عليها عليك ان تحفظها في .bash\_profile. باستخدام محرر الكتابة emacs، أتبع الخطوات التالية:
    - & .bash\_profile emacs OR & .bash\_profile emacs (يعتمد على النظام المستعمل)
    - alias dir="ls -l" تكتب في نهاية الملف ثم احفظ الملف
- 7 pwd : يستخدم لمعرفة الموقع الذي انت فيه تحت "/"
- 8 cd : يستخدم للتنقل بين الملفات
  - للإنتقال الى الأعلى
  - cd ..
  - للإنتقال الى موقع المستخدم الشخصي (Home)
  - cd ~

- للإنتقال الى الملف المقصود اسفل منك
  - للإنتقال الى اي ملف في النظام، عليك تحديد المسار
  - 9- mkdir (make directory) : لإنشاء مجلد وصيغته العامة:
  - 10- برمجة الأوامر (shell programming): (سنعطي بعض الأمثلة )
  - لإنشاء العديد من المجلدات دفعة واحدة .اتبع مايلي:
- ```

cd <folder name>
cd <folder path>
mkdir <filename>

for i in 1 2 3 4
> do
> mkdir $ i lin
> done

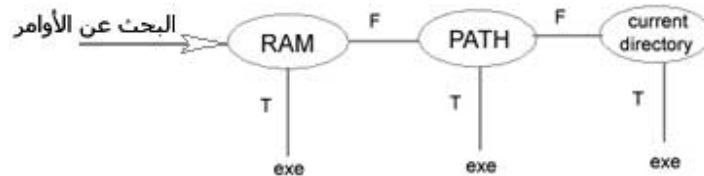
ls

1lin 2lin 3lin 4lin

```
- لفحص الناتج:
  - وستجد هذه المجلدات
  - 11- "export PSI=".....": تستخدم لتغيير شكل بداية سطر الأوامر (prompt)
  - 12- rm : تستخدم لحذف الملفات او المجلدات
  - 13- fdisk /mbr : هذا الأمر يستخدم على نظام "DOS" لإلغاء الصفحة التي تظهر عند بداية تشغيل الحاسوب التي تستعمل لإختيار نظام التشغيل الذي تريده، وذلك عند ازالة نظام "لينكس" (uninstall)

### معلومات متنوعة (من أسئلة الطلاب المشاركين)

- تندرج جميع ملفات التعريف (configration files) مثل: كلمات المرور (password) تحت /etc ويعد اهم مجلد في النظام كله، وهو مقصد قرصنة الحاسوب الأول
- PATH: يتم من خلاله تحديد المسار الذي يسلكه النظام للوصول الى الملفات(البرامج) المسؤولة عن عمل الأوامر (وهي في الغالب تحت /bin )
- يتم البحث كما في الخطط:



- عند اطلاق النظام يتم تحميل RAM بعض الأوامر مثل: cd, path, alias ،في هذه الحالة يتم تحميل برنامج الأمر وليس اسم(link) له كما هي العادة
- فإذا لم يوجد الأمر في RAM يتم البحث في PATH
- لعرض PATH في نظام "لينكس":
- echo \$PATH
- لعرض PATH في نظام "DOS":
- c:\ path
- لتغيير PATH في نظام "لينكس":
- PATH =:/..... : /.....
- لتغيير PATH (over write) في نظام "DOS":
- path=c:\.... ; c:\....
- لإضافة PATH دون تغيير الأول (append) في نظام "لينكس":
- PATH =\$PATH:/
- لإضافة PATH دون تغيير الأول (append) في نظام "DOS":
- path=% PATH % ; c:\....

- لتغيير (over write) PATH أو اضافة في نظام "لينكس" عن طريق ملف: `.profile` OR `.bash_profile` (يعتمد على النظام المستعمل)
- لتغيير (over write) PATH أو اضافة في نظام "DOS" عن طريق ملف: `autoexec.bat`

الدرس الثالث: جمع و ضغط و فك الملفات

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

### معلومات عامة

- Kernel: هو البرنامج الرئيسي الذي يتعامل مباشرة مع الأجهزة (hardware) ، وكلما كان أفضل كان النظام (operating system) أفضل
- نظام windows لديه أسوأ kernel
- نظام Novell لديه أفضل kernel

### بعض الأوامر المفيدة (Command line)

-1 tar : يستخدم لجمع الملفات في ملف واحد ، كما يستخدم في ضغط الملفات

- لجمع الملفات:  
tar cvf filename.tar <the files you want to compine>
- لضغط الملفات:  
tar cvfz filename.tgz <the files you want to compine & compress>  
OR tar cvfz filename.tar.gz <the files you want to compine & compress>

- أمثلة:
- لجمع الملفات داخل الموقع الشخصي المستخدم:  
tar cvf test.tar /home /\*
- لضغط الملفات داخل الموقع الشخصي المستخدم:  
tar cvfz test.tar.gz /home /\*

- سيتم وضع الملف في المكان الذي انت فيه (current directory)
- إذا اردت وضعه في مكان معين عليك تحديد المسار (path)  
مثال:
- tar cvf /etc/test.tar /home /\*  
لضغط الملفات التي تم جمعها دون ضغط ( filename.tar ) . استخدم:
- gzip <test.tar> test.tgz  
ل فك ضغط الملفات المجمعة و المضغوطة ( filename.tgz ) . استخدم:
- gunzip <test.tgz>
- ل فك ضغط الملفات المجمعة و المضغوطة ( filename.tgz ) الى ملفات مجمعة فقط غير مضغوطة ( filename.tar ) استخدم:  
gunzip <test.tgz> <test.tar>

-2 su : يستخدم لتغيير المستخدم الى مستخدم آخر (user) او الى (root)

- لتغيير المستخدم الى (root):  
su
- لتغيير المستخدم الى مستخدم آخر (user):  
su <user name>
- ثم يطلب كلمة المرور (password)

-3 ctrl + Alt + F1 : يستخدم لتحويل النظام من العرض عن طريق النوافذ (window display) الى العرض عن طريق سطر الأوامر (shell display) كما هو الحال في "DOS"

- يمكن ان تفتح العديد من (shell display) بتغيير F1 الى (F2 , F3 , F4 , ..... , F12) باستثناء F7 الذي يستخدم للعودة الى (window display)
- يمكن العودة بالضغط على:  
Alt + F7



- للتغيير من احد المستخدمين الى آخر لنفس shell . استخدم: (change shell) chsh
- ثم يطلب كلمة المرور (password)
- 4 chmod: يستخدم لتغيير خاصية السماح لفتح وقراءة (read) الملف او تغيير المحتوى (write) او تشغيل اي برنامج (execute)
- يحتاج الى ان تعرف ان :  $r=4, w=2, x=1$  حيث:  $r=$  read ,  $w=$  write ,  $x=$  execute
- أمثلة:
  - لتغيير الملف test الى ان يكون متاح للجميع (read + write + execute):  
chmod 777 test
  - حيث ان 7 الأولى هي للمستخدم ( $1 + 2 + 4 = 7$ ) والثانية للمجموعة (group) والثالثة للبقية (other)
- 5 chown: يستخدم لتغيير مالك الملف
- 6 touch: يستخدم لإنشاء ملف فارغ
- 7 whereis: يستخدم للبحث عن اي ملف او برنامج ومنها which. خاص بـ"لينكس" دون "يونكس"
- 8 لإغلاق النظام استخدم:
  - shutdown -h : يستخدم لعلق النظام مثل (halt)
  - shutdown -r : يستخدم لإعادة تشغيل النظام (reboot)
  - shutdown "message" -t <time> : يستخدم لعلق النظام بعد الوقت الذي تحدده انت بالدقائق (time) بعد ان يرسل رسالة الى جميع المستخدمين (في حال وجود شبكة (network))

الدرس الرابع: أوامر متنوعة مع البرمجة عن طريق سطر الأوامر (Shell programming)

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

### معلومات عامة

- للحصول على معلومات شاملة عن الملفات استخدم (ls -l)، وسيظهر لك خيارات الملف ( + write + read (execute) (راجع الدرس الثالث عند الأمر chmod). مثال:  
- (rw) (rwx) (rwx) <File Name> <File user> <File size>
- المجموعة الأولى خاصة لمستخدم الملف (user)، والثانية للمجموعة التي ينتمي إليها (group)، والثالثة لباقي المستخدمين (others)
- يبدأ عرض المعلومات بـ (-) وهو كناية عن أنه ملف وليس مجلد (Folder)، وهناك رموز أخرى مثل:
  - d : كناية أنه مجلد (Directory)
  - c : كناية أنه اسم جهاز موصول (Character device) مثل الشاشة (Monitor)
  - b : كناية أنه ملف محجوب (Block)
  - l : كناية أنه ملف وصل ورابطة (Linked File)
- كلمات المرور (Password) توجد في /etc/shadow

### بعض الأوامر المفيدة (Command line)

- 14- grep: يستخدم للبحث عن أسماء الملفات أو أسماء داخل هذه الملفات، يشابه في عمله الأمر Find في نظام "DOS" أو Search في نظام Windows
- الصيغة العامة:  
grep -<option> <Name wanted to be searched> <Where to find this word>
  - مثال:  
للبحث عن for في الملف passwd  
grep -i for /etc/passwd
  - بعض الخيارات (Options):
    - i : لإهمال تحري الحروف الكبيرة (Capital letter) والحروف الصغيرة (Small letter) بمعنى أن FoR=for
    - c : لعد الكلمات
    - n : لعرض رقم السطر الذي يحتوي على الكلمة
    - v : لعرض رقم السطر الذي لا يحتوي على الكلمة
    - w : لعرض نفس الكلمة بالضبط
  - للبحث عن الحرف الأول من الأسماء فقط:  
grep "^ <Letter>"
  - للبحث عن الحرف الأخير من الأسماء فقط:  
grep "- <Letter>"
  - للبحث عن مجموعة من الأسماء حروفها الأولى تتراوح من a الى d مثلا:  
grep "[a-d]"
  - للبحث عن مجموعة من الأسماء حروفها الأولى لا تتراوح من a الى d مثلا:  
grep "[^a-d]"  
OR grep -v "[a-d]"
- 15- info: يستخدم للحصول على معلومات عن الأوامر المتوفرة في النظام، يشابه في عمله الأمر man
- مثال:  
info ls
  - سيعرض المعلومات التوفرة للأمر ls والخيارات المتاحة

معلومات متنوعة (من أسئلة الطلاب المشاركين)

- Shell coding : البرمجة عن طريق سطر الأوامر

- **المثال الأول:**

- قم بإنشاء ملف (عن طريق محرر الكتابة emacs مثلا):

```
emacs <File Name> &
OR touch <File Name>
```

- سنعطي هنا بعض الأمثلة وما يكافئها بلغة C/C++  
▪ المثال الأول:

```
if [ $num1 -eq $num2 ]
then
:
:
:
else
:
:
:
fi
```

- ويعادلها بلغة C/C++:

```
if(num1==num2)
<statement> ;
else
<statement> ;
```

- فتكون الفروق كالتالي:

- -eq تعادل ==
- strcmp تعادل =
- -le تعادل <=
- -ge تعادل >=

- المثال الثاني:

Consider nu = 1

```
while [ $nu -le 100 ]
>do
>mkdir $nu
>nu='exp $nu + 1'
>done
```

- سيقوم البرنامج بإنشاء 100 مجلد (Directories)، وأسمائها تبدأ من 1 الى 100
- فتكون الفروق كالتالي:
- nu='exp \$nu + 1' تعادل nu++

- **المثال الثاني:**

```
for i in `cat_file`
>do
>touch $i
>done
```

- شرح المثال: هناك ملف باسم cat\_file يحتوي على عدد من الأسطر المكتوبة (قم بإنشائه أنت!). قم بكتابة هذه الأسطر على shell وسيقوم بإنشاء عدد من الملفات مساوية لعدد الأسطر في ملف cat\_file وكل سطر هو عبارة عن اسم أحد هذه الملفات المنشأة ، فلو كان الملف يحتوي على التالي:

```
1
2
3
4
```


- ثم قم بكتابة البرنامج، ثم استخدم ls ، ستجد هذه الملفات:

1 2 3 4

الدرس الخامس: تعلم برنامج "Xdialog"

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

#### معلومات عامة

- Xdialog: برنامج يستخدم لتحويل البرامج التي تعمل على سطر الأوامر الى برامج لها واجهة ونوافذ ( Graphic interface ), يشابه في عمله برنامج "Visual C++" على نظام "Windows"
- قامت مجموعة " لينكس " بعمل كتيب لتعلم البرنامج "Xdialog documentation" مع بعض الأمثلة التوضيحية
- اللغة المستخدمة في هذا الكتيب هي اللغة الإنجليزية "English"
- للحصول على كتيب التعليمات بالإضافة الى البرنامج نفسه اضغط هنا 
- الملف المضغوط Xdialog.zip يحتوي على:
  - Xdialog.pdf : كتيب التعليمات يعمل على نظام Windows
  - examples.tgz : أمثلة للعديد من التطبيقات مبتدئة بالأسهل فالأسهل يعمل على نظام Linux
  - Xdialog-2.0.4.orig.tar.gz : الذي يحتوي على البرنامج يعمل على نظام Linux

إعداد أعضاء مجموعة "لينكس" بإشراف الدكتور أحمد بنطية  
مؤسس مجموعة لينكس بكلية اتصالات للهندسة  
دولة الإمارات العربية المتحدة  
الشارقة

الدرس السادس: أوامر متنوعة مع أمثلة إضافية عن "Shell programming"

\*ملاحظة: الأوامر تم اختبارها والعمل بها على النظام (Linux mandrake 9)

### معلومات عامة

- للحصول على معلومات شاملة عن أي أمر (command) استخدم الأمر man متبوعاً باسم الأمر  
مثال:  
man grep
- سيعطيك معلومات شاملة عن الأمر grep
- هذه أفضل الطرق لتعلم أوامر نظام "Linux"

### بعض الأوامر المفيدة (Command line)

16- cut: يستخدم لتقطيع الجمل والحصول على الكلمات المطلوبة

- الصيغة العامة:  
cut -f<123.....> -<Option> <Where to find this field>
- مثال:  
قم بكتابة هذا الأمر:  
more /etc/passwd
- ستجد مثلاً أسماء وحروف وارقام مفصولة بالرمز (: مثل):  
1345 : fea345 : ..... : .....
- ثم كتبت:  
cut -f1 -d: /etc/passwd
- فإن الناتج سيكون الكلمات التي في الخانة الأولى (في حالتنا: 1345) على التفعيل الآتي:  
f : بمعنى مجال الكلمة (field) وهي الأماكن المفصولة ب":"  
f1 : أي المجال الأول الذي يحتوي على الكلمات  
d : ما سيأتي بعده هو الحد الفاصل بين المجالات (في حالتنا الحد الفاصل هو":")
- إن أردت المجالان الأول والثاني دفعة واحدة:  
cut -f1-2 -d: /etc/passwd
- إن أردت المجالان الثاني من الأخير(أي قبل الأخير):  
cut -f1- -d: /etc/passwd

### معلومات متنوعة (من أسئلة الطلاب المشاركين)

- Shell coding : البرمجة عن طريق سطر الأوامر (أمثلة إضافية ، انظر الدرس الرابع)
- المثال الأول:  
قم بإنشاء ملف (عن طريق محرر الكتابة emacs مثلاً):  
emacs <File Name> &  
OR touch <File Name>
- سنعطيك هنا المثال وما يكافئه بلغة C/C++  
for i in 1 2 3  
>do  
>echo \$i >> abc  
>done
- سيقوم البرنامج بكتابة التالي:  
1  
2  
3
- وبلغة C/C++:  
int i ;

```
for(i=1 ; i<=3 ; i++)  
    fprintf(File pointer,"%d",i);
```

▪ فتكون الفروق كالتالي:

- fprintf تعادل echo
- %d تعادل \$i