# **Addis Ababa University**

# **Addis Ababa Institute of Technology**

# **Operating Systems**

## **LAB 01**

Objective: Familiarize with the Linux environment

* Basic Linux commands
* Basic shell scripting

# **Basic Linux commands**

**ls :** this command displays the files and directories.

**man [COMMAND NAME]:** provides help on how to use a given command.

**[COMMAND NAME] -- help :** provides help on how to use a given command.

Example

**man ls**

**ls -- help**

**Exit codes in Linux :**  **echo $?**

## **Exercise 1 – File/Directory Listing**

* Display the contents of the current directory using **ls** command
* Try options **–a** or **–l** or **–i** or –R
* Use clear command to clear the screen
* Experiment with ls command by trying different options. (use **man ls** for more information)

## **Exercise 2 –Directory Commands**

* Use the **pwd** command to display the current working directory
* Create a new directory called OS using **mkdir** command
* Use the **cp** command to copy a file from the current directory to OS
* Use the **mv** command to change the name of OS to OSLAB
* Display contents of OS; type **ls OSLAB**
* Use **cd** command to change the current directory to OSLAB; **cd OSLAB**
* Use **ls** to display contents
* Change to the root directory (**cd /**)
* Change to the home directory of the current user (**cd ~**)
* Remove the OSLAB directory; **rmdir OSLAB.** Why does it fail to remove?
  + First remove the file that OSLAB contains using **rm** **OSLAB/{Filename}**.
  + Then use **rmdir OSLAB.**

## **Exercise 3 – Study the function of the following commands**

* date
* who -H
* tty
* cal
* head
* tail
* cat
* more

|  |  |
| --- | --- |
| **Expression** | **Description** |
| ^ | Start of string e.g. **(grep ^a filename)** |
| $ | End of string e.g. **(grep a$ filename)** |

* grep
* sort
* top
* output redirection (**>**) and pipe (**|**)

## **Exercise 4 – Shell Programming**

### **Usage of variable**

* Type the following using a text editor (gedit)

|  |  |  |
| --- | --- | --- |
|  | **Permission Type** | **symbol** |
| **0** | **No Permission** | **- - -** |
| **1** | **Execute** | **- - x** |
| **2** | **Write** | **- w -** |
| **3** | **Write + Execute** | **- wx** |
| **4** | **Read** | **r - -** |
| **5** | **Read + Execute** | **r - x** |
| **6** | **Read + Write** | **r w -** |
| **7** | **Read + Write + Execute** | **r w x** |

|  |
| --- |
| **echo What is your name?**  **read name**  **echo Hi $name** |

* Save the file as 1.sh
* Run the script on a terminal; **./1.sh**
* Change the file mode bits. **chmod 777 1.sh**
* Run the script again; **./1.sh**

### **Arithmetic operation**

|  |  |  |
| --- | --- | --- |
| **echo Enter X:**  **read x**  **echo Enter Y:**  **read y**  **echo Method 1**  **echo $x + $y = $[x+y]**  **echo $x - $y = $[x+y]**  **echo $x \\* $y = $[x\*y]**  **echo $x / $y = $[x/y]**  **echo Method 2**  **echo $x + $y = $((x+y))**  **echo $x - $y = $((x+y))**  **echo $x \\* $y = $((x\*y))**  **echo $x / $y = $((x/y))** |  | **echo ---SUM USING FOR---**  **for((i=0;i<=x;i++))**  **do**  **s=$[s+i]**  **done**  **echo sum 1..$x = $s**  **echo ODD/EVEN NUMBERS 1..10**  **for((k=1;k<=10;k++))**  **do**  **if((k%2 == 0));then**  **echo $k is even**  **else**  **echo $k is odd**  **fi**  **done** |

1. Write a shell script that adds numbers 1 to N (use while loop)
2. Write a shell script that calculates the factorial of N

### **Writing perl script**

|  |
| --- |
| **#!/usr/bin/perl**  **print("Welcome to Perl Addition\n");**  **print("A: ");**  **$A = <STDIN>;**  **print("B: ");**  **$B = <STDIN>;**  **$SUM = $A + $B;**  **$DIF = $A - $B;**  **$MUL = $A \* $B;**  **$DIV = $A / $B;**  **print("sum = $SUM\n");**  **print("diff = $DIF\n");**  **print("prod = $MUL\n");**  **print("div = $DIV\n");** |

* Save the file as a.pl
* Run the script on a terminal; **perl a.pl**

|  |  |
| --- | --- |
| **Expression** | **Description** |
| . | Any character |
| ^ | Start of string |
| $ | End of string |
| \* | Zero or times of preceding string |
| \ | Represent special character |
|  |  |
| ? | Exactly one character |
| () | Groups regular expressions |
| {n} | preceding character appearing n times |
| {n,m} | preceding character appearing n times but not more than m |
| {n,} | preceding character appearing n times or more |
|  |  |
| \+ | One or more occurrence of previous character |
| \? | Zero or more occurrence of previous character |

**Examples**

**echo {0..10}**

**echo {a..z}**

**echo {aa, bb, cc, dd}**

**echo a{0..9}b**

**cat filename | grep –E p\{2}** searches for p appearing exactly two times