Student Name: ______________________________ **Lab Section:** _______

/3
marks

Disks, Partitions, and File Systems - Part 2 of 2

1 Due Date - Upload to Blackboard by 8:30am Monday April 9, 2012

Submit the completed lab to Blackboard following the Rules for submitting Online Labs and Assignments.
You must upload **two** files for this submission. You must always upload **both** files when you submit. Both.

2 Commands, topics, and features covered

Use the on-line help (**man** command) for the commands listed below for more information.
The Class Notes also cover the use of these commands.

- **`fdisk`** – to display, create, delete, and manage partitions; option **`-l`** is very useful
- **`mkfs`** – create a file system on a device, usually a hard disk partition.
- **`file`** – determine what kind of thing a pathname is. Can show partition file system types using option **`-s`** and will follow (dereference) symbolic links using option **`-L`** (upper case)
- **`mkswap`** – initialize a partition for use as a Linux swap partition.
- **`swapon`** – tell the Linux kernel to use an initialized swap partition.
- **`mount`** – mount a file system existing on some device into the main file system tree.
- **`umount`** – detach (unmount) a mounted file system (e.g. that was mounted with **`mount`**).
- **`df`** – show mounted partitions and amount of used/free space
- **`du`** – recursively display disk usage in directories
- **`eject`** – unmount and eject a CDROM

3 Correct user, command lines, and command output

- Most of this lab is done as the **root** user. Your prompt will tell you if you are the **root** user.
- Some answer blanks require you to enter **command lines**. Do **not** include the shell **prompt** with your command lines. Give only the part of the command line that you would type yourself.
- Make sure you know the difference between a command **line** (which is what you type into the shell) and command **output** (which is what the command displays on your screen).

4 Use your 1 GB disk and five partitions from the previous steps

To continue this lab, you must have successfully created these **five** (remaining) partitions on the 1GB disk:

```
# fdisk -l /dev/sdb
   Device Boot    Start        End      Blocks   Id  System
/dev/sdb1             1         26      208813+  83  Linux
/dev/sdb2            27         40      112455   83  Linux
/dev/sdb3            41        119      634567+   5  Extended
/dev/sdb5            67         80      112423+  82  Linux swap / Solaris
/dev/sdb6            81        119      313236    7  HPFS/NTFS
```

Do not proceed until you have the above five partitions created. The sizes may vary *slightly*. The System ID must match. The end of the extended partition must be **less than** cylinder 130. (130 is the end of the disk). All file system commands in this lab referring to a hard disk will **use one of the above partitions**.

1. Create a **back-up** of your working system now using the VMware **snapshot** function. You can return to this back-up if anything goes wrong during this lab, or if you want to repeat the lab for practice.
2. Obtain a **root** (super-user) shell prompt so that you'll have the required **privilege** level to run the **file system maintenance** commands. Run all maintenance commands as **root**.

5 *Creating file systems using* `mkfs`

To create a file system on an *existing* **partition** or device, use one of the **`mkfs`** commands. Syntax:

➢ **`mkfs -t`** ***`filesystem_type device_name`***

For example: **`mkfs -t ext3 /dev/sdb9`**
The *device_name* is usually the name of an *existing* **partition**, not the name of the whole disk.

1. Find the device names on your 1GB disk of the **two** partitions that have partition type (System ID) of "**Linux**". Create a Linux type **ext3** file system on the *first* Linux partition. Record the exact command line you used, and the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:
   ____
   ____

2. Create an **ext4** file system on the *second* partition that has type "**Linux**". Record the exact command line you used, and the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:
   ____
   ____

3. Create an **ntfs** file system on the only "**HPFS/NTFS**" partition. Record the exact command line you used, and the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:
   ____
   ____
   ____
   ____

4. Take a VMware back-up snapshot now and name it something like "**done_mkfs**".

6 *Mounting & Unmounting a Linux File system using* `mount`

Creating a **file system** does not automatically connect that file system to your Linux system. To connect an existing file system to your Linux file system so that you can access it, use the **`mount`** command. Syntax:

➢ **`mount`** ***`[-t filesystem_type] device_name mount_point`***

For example: **`mount -t ext4 /dev/sdb9 /mnt/foobar`**
The *device_name* is usually the name of an existing **partition**, not the name of the whole disk.
The *mount_point* must already exist. Let me repeat that: **THE MOUNT POINT MUST EXIST**.

- While completing this next section of the lab, use **`mount`** without any arguments frequently to confirm your actions and ensure that mounting and unmounting were successful. This is how you see what is mounted and what type of thing each is.

1. List all the partitions on your system using: **`fdisk -l`**
   You may see two error messages about **`/dev/dm-0`** and **`/dev/dm-1`** that you can ignore.
   Record the sizes of Disk **`/dev/sda`** and Disk **`/dev/sdb`**: ________________________________
2. List all the mounted file systems using: **`mount`**
   What partition device name is mounted on the **`/boot`** directory? ________________________

6.1 Mount three file systems

1. Create three directories named **`/mnt/ext3`**, **`/mnt/ext4`**, and **`/mnt/ntfs`** to use as mount points for the three file systems you created at the end of Section 3. After creating them, record the output of:

   `ls -id /mnt/ext3 /mnt/ext4 /mnt/ntfs`

   ____

2. Use three **mount** commands to mount the three file systems you created previously, each mounted on its own directory from above. (Recall that each file system was created with a particular type. Match the file system type with the directory name.) Record the three **mount** commands you used to do this here:

   ____

   ____

   ____

3. Use **mount** without any arguments to verify that you have three new mounted file systems. Each file system type should match the directory name on which it is mounted. Each file system should be mounted only **once**. (If you have duplicate entries, unmount them using the **umount** command.)
4. Record again the (different!) output of: **ls -id /mnt/ext3 /mnt/ext4 /mnt/ntfs**

   ____

5. Leave the three file systems mounted. Take a VMware back-up Snapshot now and name it something like "**done_3mount**".

7 Preparing a Swap Partition on a hard drive using mkswap and swapon

Linux systems usually have an entire partition devoted to **swap area** to enable **virtual memory**. The partition must be initialized with **mkswap** and then given to the Linux kernel using **swapon**. Syntax:

- **mkswap *[options] device_name***
- **swapon *[options] [device_name]***

The ***device_name*** is the **pathname** (usually **absolute**) of the existing swap partition that will be used.

1. Enter **three** commands: Initialize the **Linux swap** partition on your new 1 GB disk; tell the kernel to use it; display the active swap partitions. Record the **three** command lines used to do these **three** actions:

   ____

   ____

   ____

2. Record the **output** of "**file -s *device_name***" for the ***device name*** you used:

   ____

3. Leave the new swap area connected. Take a VMware back-up Snapshot now and name it something like "**done_swap**".

8 Creating a Linux File system on a virtual Floppy disk and mounting/unmounting/ejecting it

8.1 Creating a VMware virtual floppy disk image file

To do some exercises, you first need to create a virtual floppy disk image file and connect it to VMware:

1. In VMware, click **VM → Settings**
2. Select **Floppy** in the **Virtual Machine Settings** window
3. Select **Use a floppy image**
4. Select the **Create...** button to create a new floppy disk image file. Name the file: **linux.flp** and select **Open** to create the file and attach it to your Fedora virtual machine. (Remember where the file is on your disk, so that you can find it again later!)
5. Back in **Virtual Machine Settings**, ensure that the floppy is **Connected**: make sure the **Device Status Connected** checkbox is checked.
6. ***DO NOT*** check the **Connect at power on** box or your Fedora machine will not boot.
7. Click **Save** to save the new **Virtual Machine Settings**.
8. At your Fedora super-user command line, verify that the floppy disk image file is connected by using:

```
fdisk -l /dev/fd0
```

The first line of output should read: **`Disk /dev/fd0: 1 MB, 1474560 bytes`**

8.2 Creating a Linux file system on a virtual floppy disk and mounting/unmounting/ejecting it

- You must have a virtual floppy disk image file connected to your Fedora machine to do this section. Use the **`fdisk`** command to verify that the floppy disk image file is connected to **`/dev/fd0`**
- Floppy disks appear in Linux with device names such as **`/dev/fd0`** (floppy disk zero) and are used **without** any partition numbers. Do **not** use **`fdisk`** to create partitions on a floppy disk. You create file systems directly on a floppy disk using **`mkfs`** by inserting a floppy disk (or connecting a virtual disk image file) and then using its Linux device name with no partition name, e.g. **`/dev/fd0`**
- Floppy disks are small enough that the default **ext2** file system type is best; do not create any journalling file systems on a floppy disk. Always use the **ext2** file system type for a Linux-style floppy disk.

1. Create a type **ext2** (the default) file system on your floppy disk. Record the exact command line you used, and the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:

   ____

   ____
2. Mount the floppy disk file system on **`/mnt/linux`** after creating it. Record the command used:

   ____
3. Use **`mount`** without any arguments to verify that you have a new **ext2** file system mounted and copy the single line of **output** related to the only floppy disk device here:

   ____
4. Copy a file to the floppy disk: **`cp -a /etc/passwd /mnt/linux/passwd`**
5. Use this command to see the Linux directory contents: **`ls -lia /mnt/linux`**
   Copy the two lines of output for the **`lost+found`** directory and the **`passwd`** file here:

   ____

   ____
6. To "eject" a virtual floppy disk, first **un-mount** it from Linux and then **un-check** the VMware **Device Status Connected** checkbox. Do **not** eject a floppy disk that is still **mounted** in your Fedora machine! Unmount it **first**, and only then disconnect it. Do this now: un-**mount** the floppy and then eject it. Verify that "**`fdisk -l`** ***`device_name`***" gives no output for the unmounted floppy device and copy here the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:

   ____

9 Creating a Microsoft FAT file system on a virtual floppy disk and mounting it

9.1 Creating a VMware virtual floppy disk image file

- Linux can mount many types of file systems, including Microsoft file systems. Microsoft machines can only mount Microsoft file systems. Let's create a Microsoft-format diskette and mount it in Linux.

1. **Create** and attach to Fedora a **new** virtual floppy disk image file named: **`windows.flp`**
   This new image file you create will **replace** your Linux floppy image on the **virtual** floppy disk drive. Make sure you have unmounted your Linux floppy image file before you create and switch images!
2. Verify that the floppy disk image file is connected as you did in an earlier section above.
   The first line of output should read: **`Disk /dev/fd0: 1 MB, 1474560 bytes`**

9.2 Creating a Microsoft FAT file system on a virtual floppy disk and mounting it

1. Create a type **vfat** file system on your Windows floppy disk. Record the exact command line you used, and the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** you used:

   ____

   ____

2. Mount the floppy disk file system on **`/mnt/vfat`** after creating it. Record the command used:

   ____

3. Use **`mount`** without any arguments to verify that you have a new **vfat** file system mounted and copy the single line of **output** related to the only floppy disk device here:

   ____

4. Copy a file to the diskette: **`cp -a /etc/passwd /mnt/vfat/passwd`**
5. Use this command to see the Linux directory contents: **`ls -lia /mnt/vfat`**
   Copy the one line of output for the new copy of the **`passwd`** file here:

   ____

6. Why is there no **`lost+found`** directory shown in the output above?

   ____

7. Leave the **vfat** floppy mounted. Take a VMware back-up Snapshot now and name it "**done_vfat**".

10 Mounting & unmounting a CDROM file system

- **IMPORTANT NOTE:** *All removable media, such as floppies, USB, external hard drives, and CD-ROMs, must not be removed from the drive, disconnected, or swapped with another disk while* ***mounted****. Removing or changing media that is* ***mounted*** *causes the data in memory and the data on the device to be out of synchronization. This can result in data loss and other severe errors. Whenever you want to switch or remove a floppy, USB, or CD-ROM,* ***unmount*** *it first, before removing or changing the medium. Always unmount first!*

1. Connecting the physical CD/DVD drive to Linux through VMware is much like selecting a floppy disk image file. In the VMware settings for your Fedora VM, under **CD/DVD (IDE)** under **Device Status**, select the checkbox beside "**Connected**", and in the same window, under "**Connection**", select "**Use a physical drive**", with **Device** set to "**Auto Detect**" (or try the other setting, too). **Save** the settings.
2. Now put a **data** CD-ROM disk in the CDROM drive (e.g. your Fedora disk, or a lab boot disk, or a Windows Install disk). Note: an Audio CD will **not** work. It must be a **data** disk.
3. Fedora may "automount" some disks (especially CDROMs) and you will need to **unmount** them before you can mount them manually using shell commands. Use the **`mount`** command to see if the CDROM device **`/dev/sr0`** has been mounted by Fedora and **unmount** the device if it has been automounted.
4. Make sure no CD is mounted on your Fedora desktop before you continue; unmount **`/dev/sr0`**
5. Linux often creates many symbolic links in the **`/dev`** directory that point to the **`sr0`** device name. Use a two-command pipeline to show **all** these symbolic links (more than one) and record the **pipeline** here:

   ____

6. Verify that Fedora can access your physical CD: **`fdisk -l`** ***`device_name`***
   You should see some warning messages, disk size information, and partition information.
7. Give the **output** of "**`file -s`** ***`device_name`***" for the ***device name*** of the CDROM:

   ____

8. Mount the CDROM file system on **`/mnt/cdrom`** after creating it. Record the command used:

   ____

9. Use **`mount`** without any arguments to verify that you have a new file system mounted from the CDROM device and copy the single line of **output** related to the only CDROM device here:

   ____

10. Try to copy a file to the CDROM: **`cp -a /etc/passwd /mnt/cdrom/passwd`** and record the error message here:

    ____
11. Leave the CDROM mounted. Take a VMware back-up Snapshot now and name it "**done_cdrom**".

11 Lab Check and Upload - `lab09marks.txt`

This is the section that tests and marks the work you did above. The Lab Check program will do the checking to make sure you got things right. Do you have three partitions, one floppy disk, and one CDROM mounted? Is the new swap area configured and in use?

Following a method similar to previous labs, download and run the **`lab09check`** program and upload **both** this ODT file and the **`lab09marks.txt`** file to Blackboard. Always upload **both** files together at the same time. Use the correct names. Do not print the huge **`lab09marks.txt`** file!

12 Command: `eject` ***`device_name`***

The **`eject`** command will unmount a given CDROM file system if its mounted, and eject the media from the drive. On our Fedora systems, the default device is the **cdrom** device, so just "**`eject`**" with no arguments will also work.

13 Ejecting and unmounting everything

After marking your lab, give the command lines to do these operations (and do them!):

1. Eject the CDROM. ____
2. Unmount the VFAT file system. _________________________________
3. Unmount the NTFS file system. _________________________________
4. Unmount the EXT4 file system. _________________________________
5. Unmount the EXT3 file system. _________________________________

14 Practice these commands

Go back to your earlier snapshots and **REPEAT** these exercises as often as necessary, until you can do this without looking at the lab instructions. Without looking at the instructions, can you do the following:

- Create a new VMware virtual disk and connect it to Fedora (e.g. create **`/dev/sdc`**).
- Create primary, extended, and logical partitions on the new disk.
- Set partition types.
- Create any type of file system inside any partition.
- Mount and unmount any file system

Can you do all the above operations without reference to any help files? Practice!