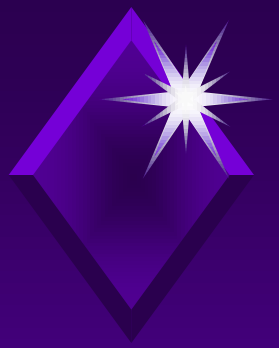
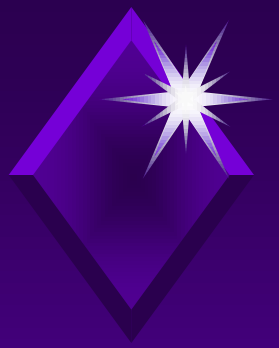


Chapter 4

Program Files, Data Files, and Subdirectories

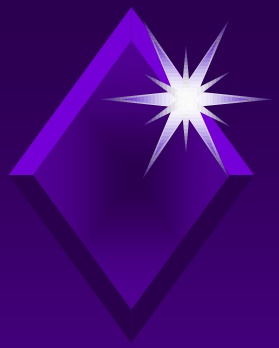


Overview

The cooperative effort between the operating system and the application program and its data files will be discussed.

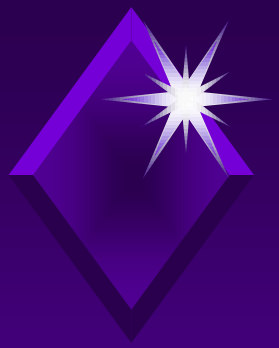
Overview

Shareware and freeware will be compared and contrasted.



Overview

Will learn to differentiate between a program file and a data file.



Overview

The hierarchical filing system of a tree-structured directory will be explained.

Overview

Will identify and use subdirectory commands to help manage files from the command prompt.

Why Use the Command Prompt Screen?

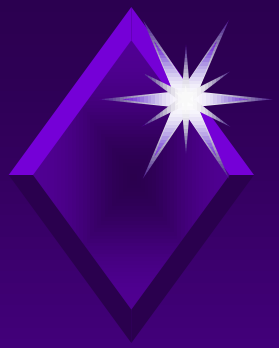
Review commands learned:

- FORMAT
- DISKCOPY
- DIR
- CLS

Why Use the Command Prompt Window?

Four major categories of application programs:

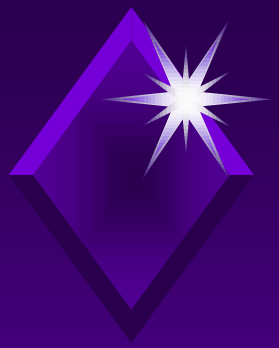
- Word processors
- Spreadsheets
- Databases
- Graphics



Why Use the Command Prompt Window?

OS is important:

- ➔ Manages the system.
- ➔ Enables the user to manage and manipulate files on disks

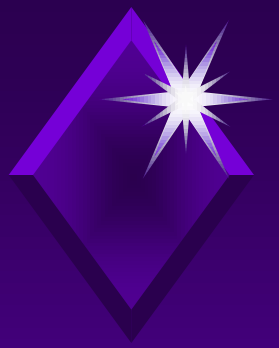


Why Use the Command Prompt Window?

The OS manages the files **not** the information put into files.

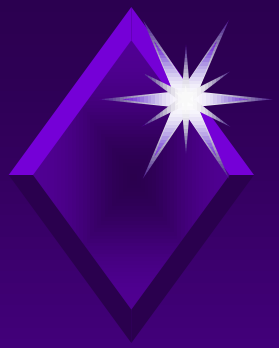
Program Files, Data Files, and the OS

- WUGXP Subdirectory contains:
  - Games which contains
    - BOG2 which contains
      - Application program called BOG



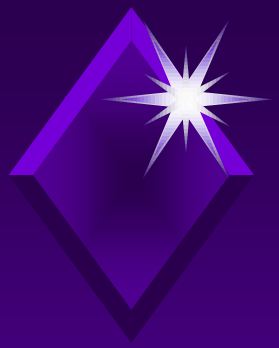
Program Files, Data Files, and the OS

Legacy software can be used by Windows OS because of downward compatibility.



Program Files, Data Files, and the OS

MS-DOS commands work in conjunction with various types of files.



Program Files, Data Files, and the OS

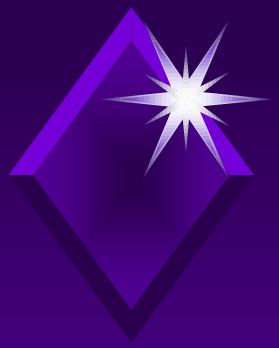
Real mode operation:

- Used for software written for DOS and early versions of Windows
- Application program interfaces with device or DOS and DOS does the work

Program Files, Data Files, and the OS

Protected mode operation:

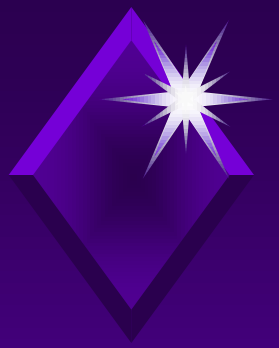
- Used for Windows software
- Application software does NOT interface with hardware
- Drivers talk to virtual device drivers



Program Files, Data Files, and the OS

Operating system:

- Loads application program into memory
- Assists in loading data file into memory
- Ensures cooperation between application program and its data files



Program Files, Data Files, and the OS

Commands:

- Are programs
- Allow user to interface with OS to manage programs and data files

Shareware

Freeware & shareware programs available from a wide variety of sources.

Shareware

Freeware:

- Software that is in the public domain

Shareware

Shareware:

- Trial version of a program

Shareware

Register shareware program to receive:

- Full version with documentation
- Update notices
- Technical support

Shareware

Appendix A lists all shareware programs with fees and addresses necessary to register them.

Activity—Using DIR to locate the BOG Program

KEY CONCEPTS:

- Use DIR command to verify BOG is on hard disk
- Function of .exe and .dat

Using Application Programs & Data Files

- DIR command acknowledges files exist.
- To use file, must load it into memory.

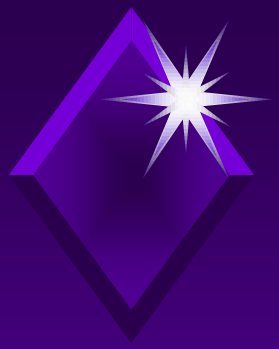
Using Application Programs & Data Files

- BOG.EXE is the application program.
- BOG.DAT is the data file.

Activity—Using Application Programs and Data Files

KEY CONCEPTS:

- Only programs can be executed
- Function of file extension
- BOG.DAT is the data file



Managing Program and Data Files at the Command Prompt

Organize programs and files into subdirectories so it will be easier to save and locate them.

Hierarchical Filing Systems or Tree-Structured Directory

Root directory:

- Created when disk is formatted
- Represented by (\) - the backslash
- Acts as an index to disk

Hierarchical Filing Systems or Tree-Structured Directory

Table 4.1 FAT16 Root Directory File Limits p. 135

Disk Size	Number of Root Directory Entries
3½-inch DS/DD disks	112
Hard disk	512

Hierarchical Filing Systems or Tree-Structured Directory

- FAT16
  - Root directory - fixed size & location on disk
- FAT32
  - Root directory free to grow as necessary
- NTFS
  - No limit to # of files/directories in root directory

Hierarchical Filing Systems or Tree-Structured Directory

Subdirectories:

- Important part of organizing disk
- Can contain subdirectories
- No limit in number of files

Hierarchical Filing Systems or Tree-Structured Directory

Subdirectory structure:

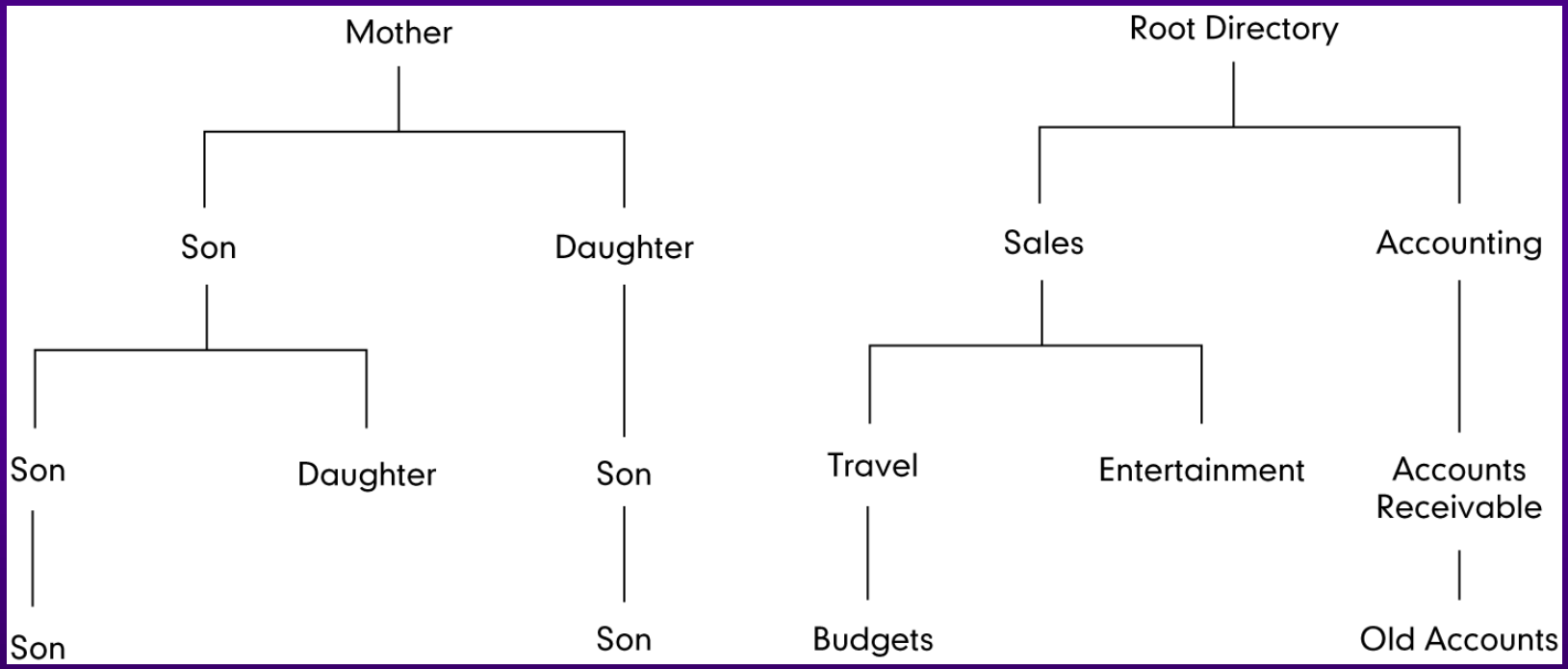
- Looks like inverted family tree -
- Root directory
  - At top of tree
  - Point of entry in hierarchical structure

Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.1 A Directory Is Like a Family Tree p. 135

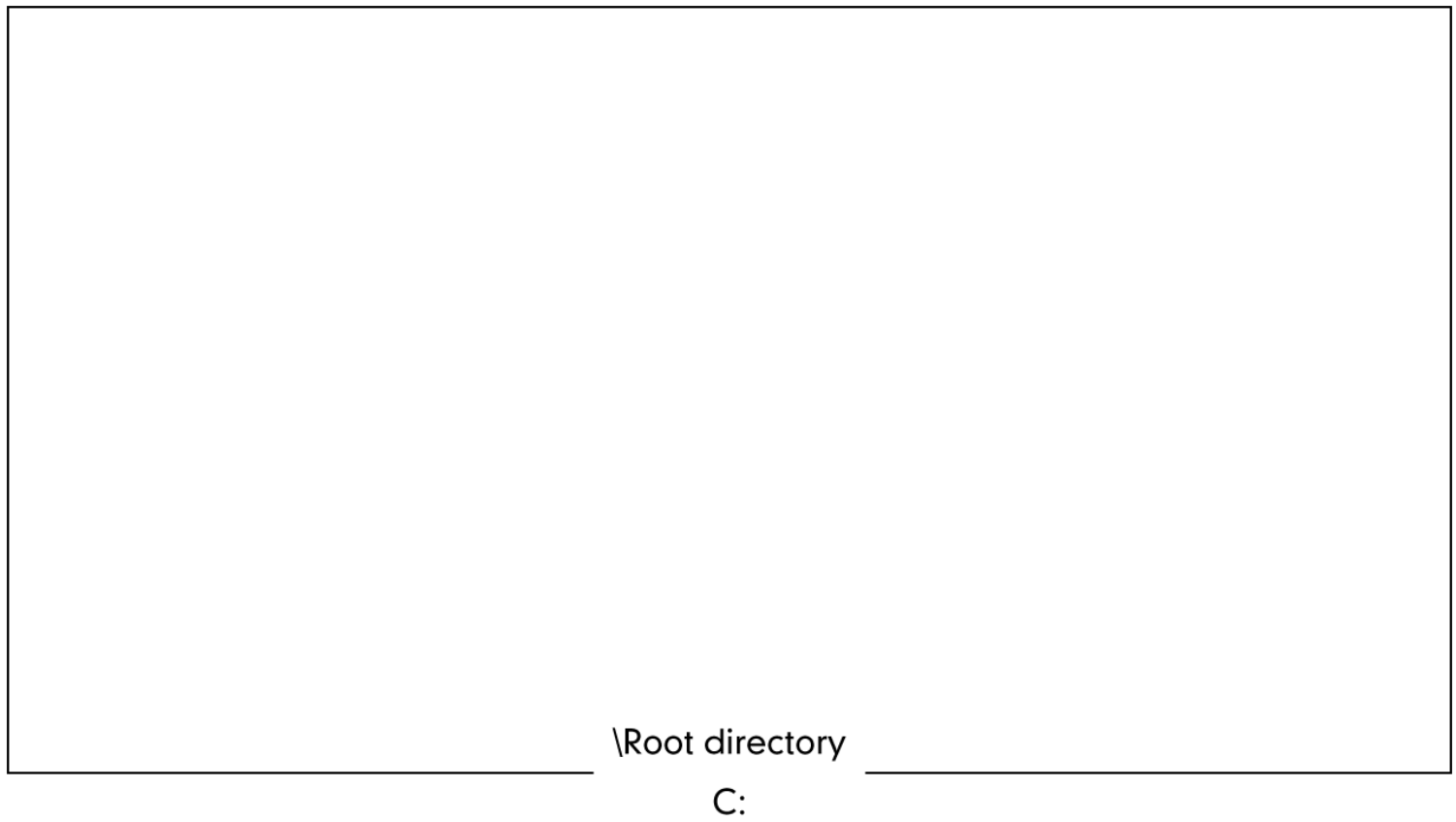
Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.2 Hierarchical Structure of a Directory p.136



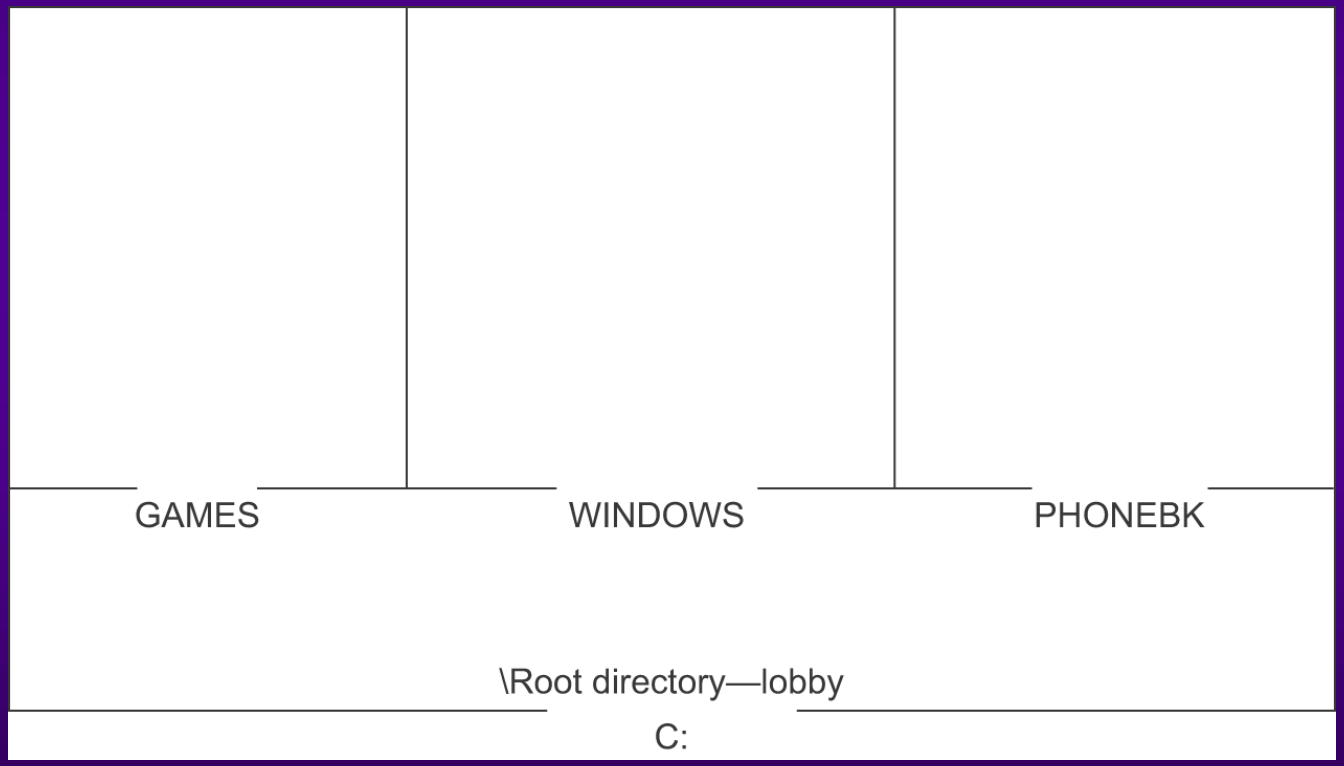
Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.3 A disk as a Building p. 136



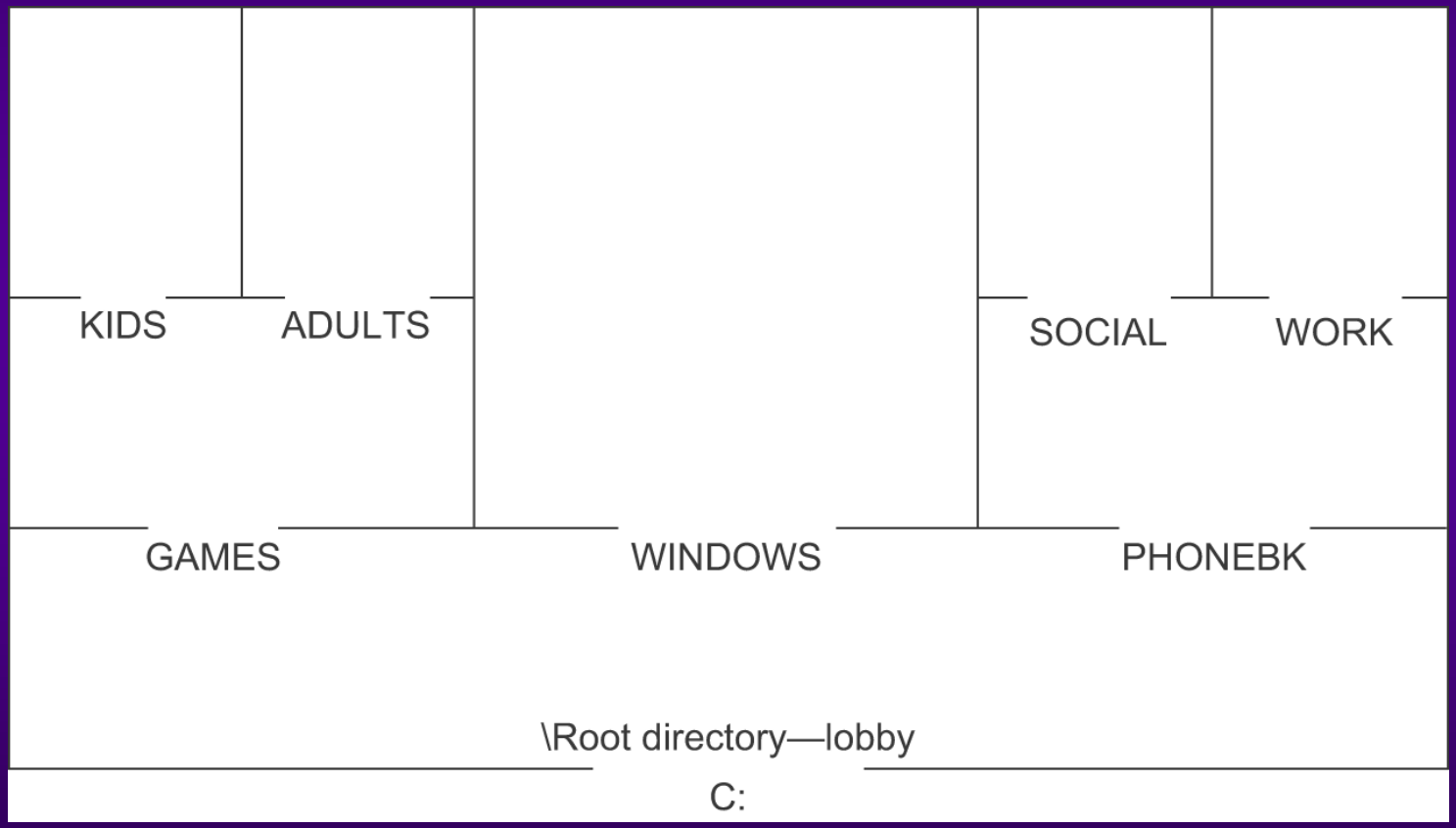
Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.4 Subdirectories as Rooms p.137



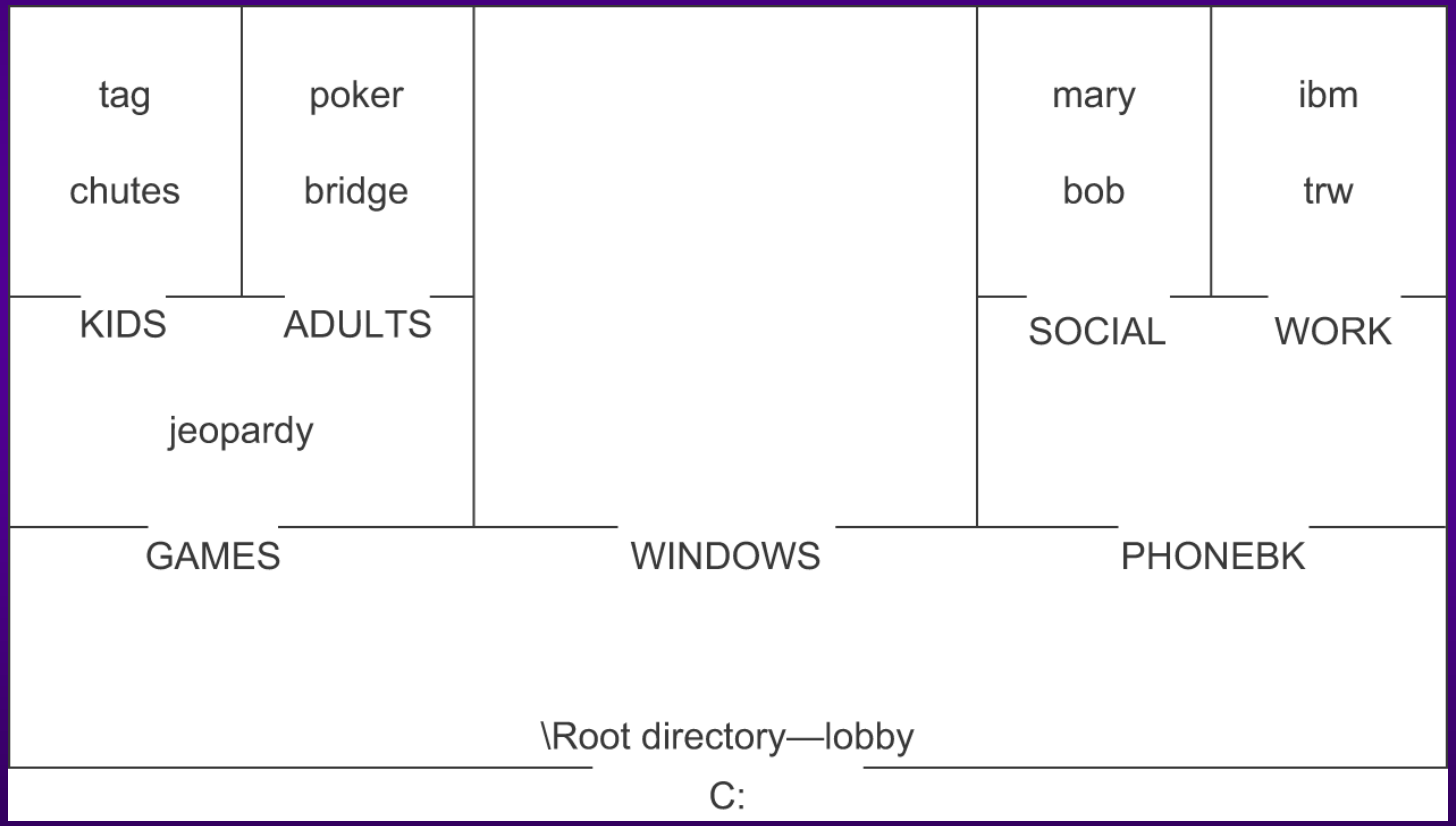
Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.5 More Subdirectories p. 137



Hierarchical Filing Systems or Tree-Structured Directory

Fig 4.6 Files in Subdirectories p. 138



Hierarchical Filing Systems or Tree-Structured Directory

General comments:

- Only one root directory - point of entry
- Directories
  - Have only one parent directory
  - Can have any # of child directories

Hierarchical Filing Systems or Tree-Structured Directory

General comments:

- Each subdirectory
  - Dependent upon structure above it
  - Knows only its parents and children
- Not changing size of structure - merely organizing it

Hierarchical Filing Systems or Tree-Structured Directory

General Comments:

- User names the subdirectories **not** the root directory
- Subdirectories
  - Follow file naming rules
  - Have special commands

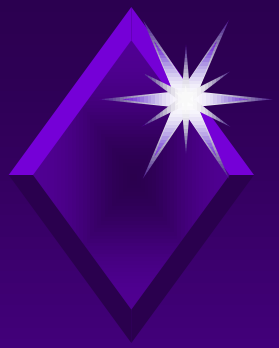
Hierarchical Filing Systems or Tree-Structured Directory

Table 4.2 Directory Management Commands p. 139

Command	Function
CHDIR or CD	Changes a directory.
MKDIR or MD	Makes or creates a directory.
RMDIR or RD	Removes or erases a directory and its subdirectories.
PATH	Defines the search paths.
PROMPT	Changes the look of the prompt to identify what subdirectory is the default.
MOVE	Allows you to rename a directory.

Creating Subdirectories

- Format disk - preparing it to hold files.
- Set up subdirectory - preparing it to hold logical group of files.



Creating Subdirectories

Subdirectory command syntax:

MKDIR [drive:] path

or

MD [drive:] path

Activity—How to Create Subdirectories

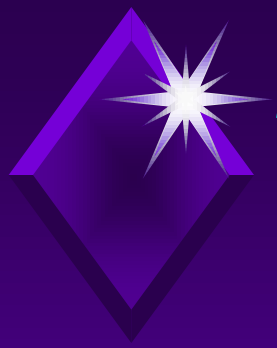
KEY CONCEPTS:

- USED /Q parameter to format disk and used shortcut to place volume label on disk
- Determine when at root directory
- Verify subdirectory created
- Attributes (D, H, S, R, A, -)
- Use DIR to display directory contents
- Subdirectory has two named subdirectories
  - Dot and double dot

The Current Directory

Operating system keeps track of:

- Default drive
- Current (default) directory for each disk

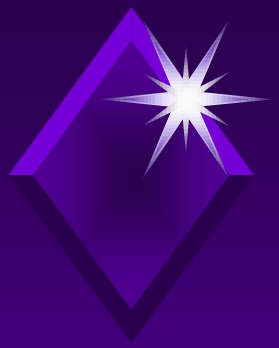


The Current Directory

CD command:

- Displays current default directory:
  - CD with no parameters
- Changes default directory:
  - Parameter after CD

    CD [drive:][path]



The Current Directory

- CD command does not use spaces as delimiters.

- Syntax
  - CD [/D] [drive:] [path]

The Current Directory

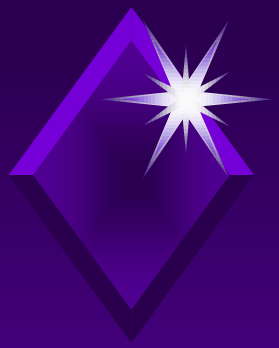
What happens if prompt is A:\> and you key in:

- CD C:\WUGXP
- C:
- CD /D C:\WUGXP

Activity—Using the CD Command

KEY CONCEPTS:

- Results of keying in CD
  - With no parameters
  - Followed by subdirectory name
- CD alone cannot be used to change drives
- Results of keying in DIR



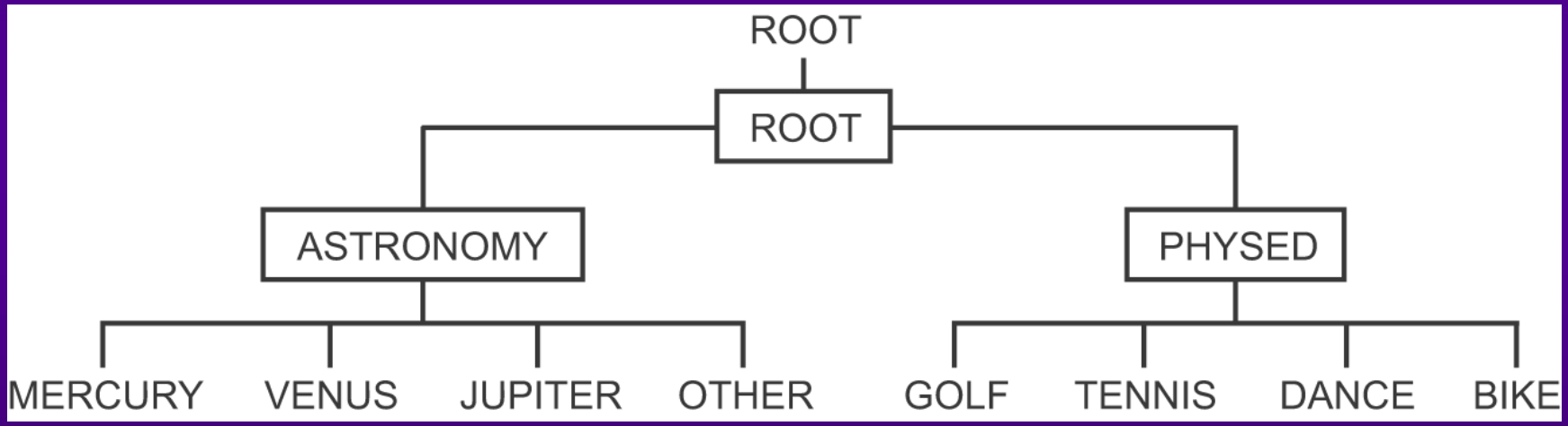
Relative and Absolute Paths

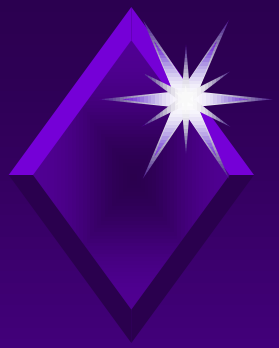
Use MD or MKDIR command to make new subdirectories.

MD [drive:] path

Relative and Absolute Paths

Fig 4.7 Directory with Subdirectories p. 147

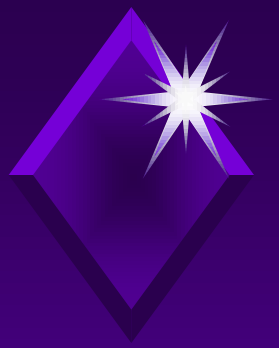




Relative and Absolute Paths

Absolute path:

- Direct route from root directory to the subdirectory of interest
- Is always absolutely correct
- Complete and total hierarchical structure



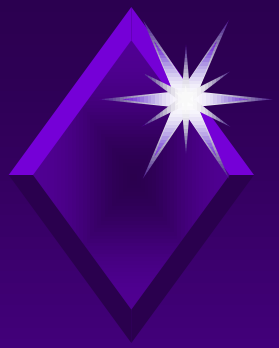
Relative and Absolute Paths

Relative path:

- Route from where you are to where you want to go
- Can move to directory above it and beneath it

Relative and Absolute Paths

- Directory:
  - Knows about files/directories within itself
  - Knows about its immediate child directory and parent directory
  - To move to another parent directory must return to the root



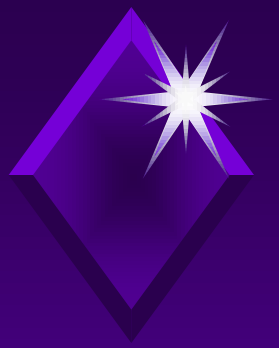
Relative and Absolute Paths

The root directory is parent or common “ancestor” of all directories.

Activity—Creating More Subdirectories

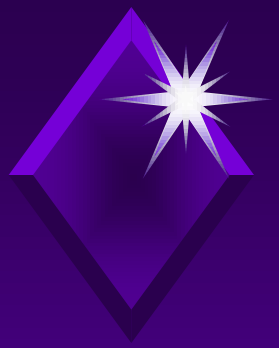
KEY CONCEPTS:

- Absolute vs. relative path
- Hierarchy
- File vs. directory
- Create subdirectories
- When and when not to use \



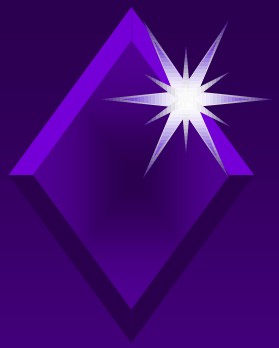
Knowing the Default Directory

It is important to know default drive and directory.



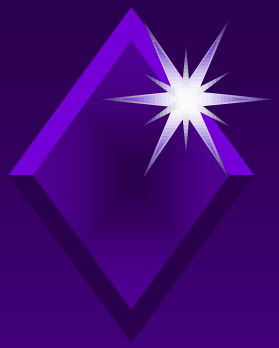
Knowing the Default Directory

- PROMPT command without parameters displays current drive and > sign.
- CD command displays default or current drive and directory.



PROMPT Command

- If no prompt specified prompt includes path and >.
- Changing way prompt is displayed does not change its function.



PROMPT Command

PROMPT command:

- Contained in CMD.EXE
- Syntax - **PROMPT [text]**
- Can include **metastrings**

PROMPT Command

PROMPT command Metastrings p. 153

Character	Description
$A	& (ampersand)
$B	| (pipe)
$C	((left parenthesis)
$D	Current date
$E	Escape code (ASCII code 27)
$F	) (right parenthesis)
$G	> (greater-than sign)
$H	Backspace (erases previous character)

PROMPT Command

PROMPT Command Metastrings p. 153

Character	Description
$A	& (Ampersand)
$B	¦ (Pipe)
$C	((Left parenthesis)
$D	Current date
$E	Escape code (ASCII code 27)
$F	) (Right parenthesis)
$G	> (Greater-than sign)
$H	Backspace (Erases previous character)
$L	< (Less-than sign)
$N	Current drive
$P	Current drive and path
$Q	= (Equal sign)
$S	(Space)
$T	Current time
$V	Windows XP version number
$_	Carriage return and linefeed
$$	$ (Dollar sign)

Activity—Changing the Prompt

KEY CONCEPTS:

- Changing appearance of prompt does not change its function
- Important to display default drive letter
- Can return prompt to default value by keying in command with no parameters

Subdirectory Markers

- Single . (one period) - specific name of current directory.
- Double . . (two periods) - specific name of parent directory of current subdirectory.

Subdirectory Markers

- Can use . . (two periods) to move up the directory.
- Cannot use shortcut symbol to move down hierarchy.

Activity—Using Subdirectory Markers

KEY CONCEPTS:

- Using . . (two periods)
- Space after CD/MD command & before backslash or directory marker is optional
- When to use relative path
- When to use absolute path
- Keying in CD. . vs. keying in CD \

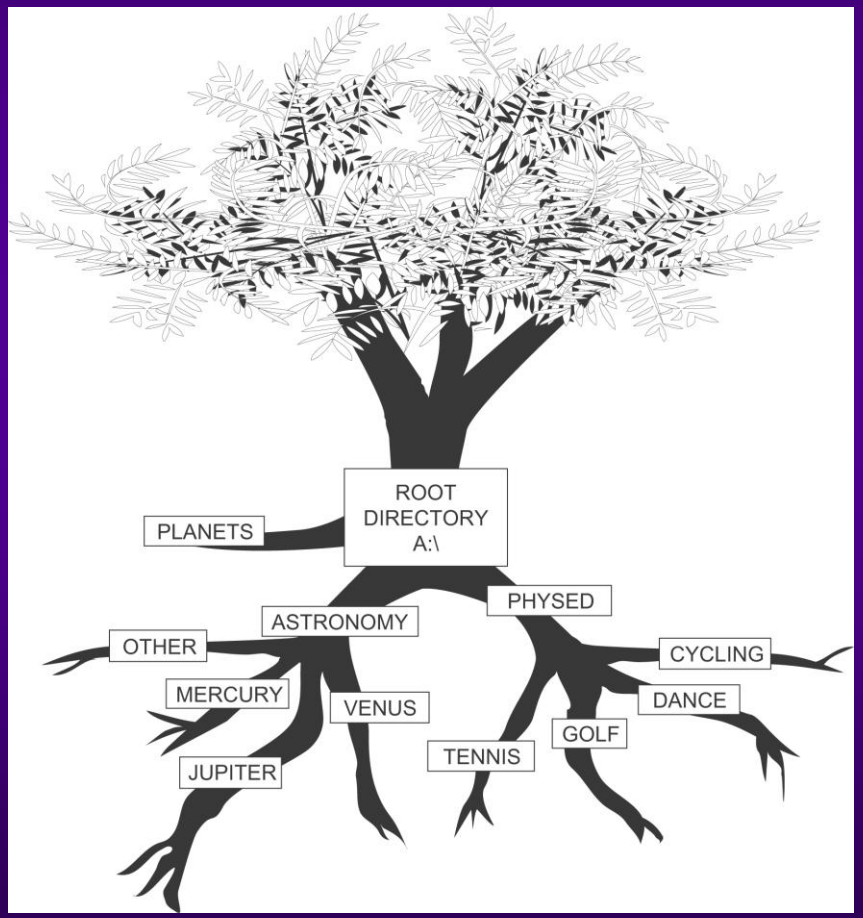
Activity—Using Subdirectory Markers

KEY CONCEPTS:

- What happens if MD \GOLF keyed in instead of MD PHYSED\GOLF
- Results of keying in CD \ or CD\

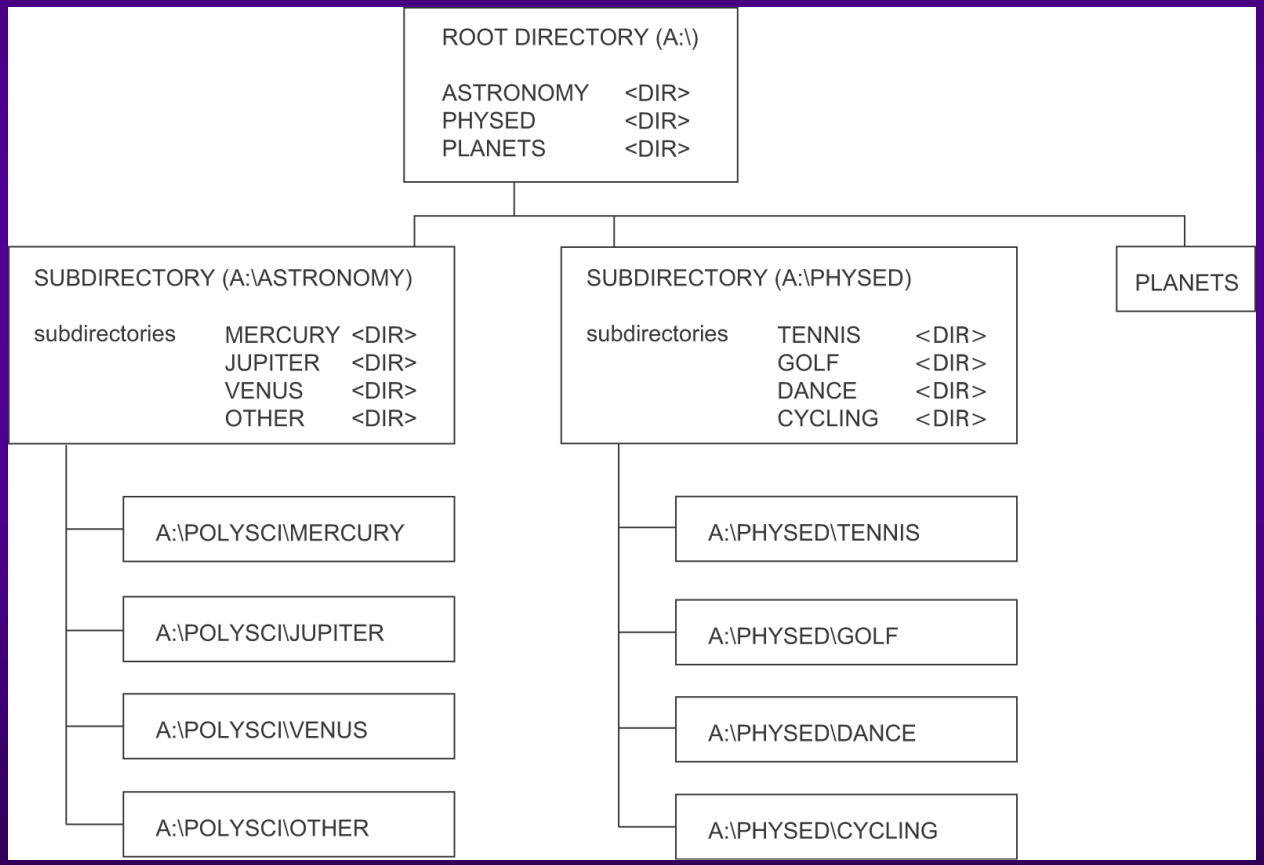
Activity—Using Subdirectory Markers

Fig 4.8 Structure of the Data Disk p. 158



Activity—Using Subdirectory Markers

Fig 4.9 Subdirectories: Another View p. 159



Changing the Names of Directories

MOVE command:

- Used to rename directory from the MS-DOS prompt
- MOVE [/Y | /-Y] [drive:] [path]
  - dirname1 direname2

Activity—Using MOVE to Rename a Directory

KEY CONCEPT:

- Give correct path name (absolute or relative) and can rename a directory from any location

Removing Directories

RD or RMDIR command:

- Used to remove directories
- Cannot remove
  - Directory containing hidden or system files
  - Directory you are in
  - Default subdirectory
  - Root directory
- Cannot use with wildcards

Removing Directories

RD [/S] /Q] [drive:] path

Without parameters:

- Removes:
  - Only empty subdirectories
  - Files one at a time - bottom up

Removing Directories

RD [/S] /Q] [drive:] path

With parameters:

- Can remove directory tree
- Can remove directory with hidden or system files
- Can traverse directory tree from top down

Activity—Using the RD Command

KEY CONCEPTS:

- Directory is type of file
- Cannot delete
  - Directory you are in
  - Default directory
  - Root directory

Activity—Using the RD Command

KEY CONCEPTS:

- Create directories - top down
- RD without parameters
  - Remove directories - bottom up
  - Removes empty directories

Deleting a Directory and Its Subdirectories

RD [/S] [/Q] [drive:] path

RD with /S parameter:

- Deletes directory and contents (subdirectories/files) with one command
- Removes a tree

Deleting a Directory and Its Subdirectories

RD with /S parameter:

- Traverse directory tree from top down
- With /Q parameter - RD runs in quiet mode
- Useful, fast, powerful, dangerous

Activity—Using RD with the /S Parameter

KEY CONCEPTS:

- RD /S
  - Removes files and subdirectories with one command
  - Useful, fast, powerful, dangerous

Using Multiple Parameters with MD and RD

Can create (MD) or remove (RD) more than one directory on the same command line.

Using Multiple Parameters with MD and RD

MD command:

- Create parent and child directory with one command.
- If parent directory does not exist, the OS will create child directories and any necessary intermediate directories.

Activity—Using Multiple Parameters with MD & RD

KEY CONCEPTS:

- Creating subdirectories with MD
- Removing subdirectories with RD
- Using /S and /Q parameters with RD

Understanding the PATH Command

Covered in this chapter:

- CD command
- Located/executed BOG
- Reviewed process of executing a program
  - .com, .exe, and .bat extensions denote executable programs
- Used MD, DIR, CD, RD PROMPT, FORMAT, DISKCOPY, and MOVE

Understanding the PATH Command

OS Search for Correct File is Limited to File Extensions in Order Listed p.168

Extension	Meaning
.COM	Command file
.EXE	Executable file
.BAT	Batch file
.CMD	Command script file
.VB	VBScript file (Visual Basic)
.VBE	VBScript Encoded Script file (Visual Basic)
.JS	JScript file (JavaScript)
.JSE	JScript Encoded Script file (JavaScript)
.WSF	Windows Script file
.WSH	Windows Script Host Settings file

Understanding the PATH Command

PATH command:

- Locates/executes executable program files
- Searches
  - Memory
  - Current directory
  - Subdirectories specified with PATH command

Understanding the PATH Command

PATH command syntax:

PATH [[drive:] path [;…] [%PATH%]]

Activity—Using the Path Command

KEY CONCEPTS:

- Use of ; following PATH command
- Location of BOG
- Elements of path separated by (;)
- No need to replace existing PATH to change it
- Returning PATH to original setting