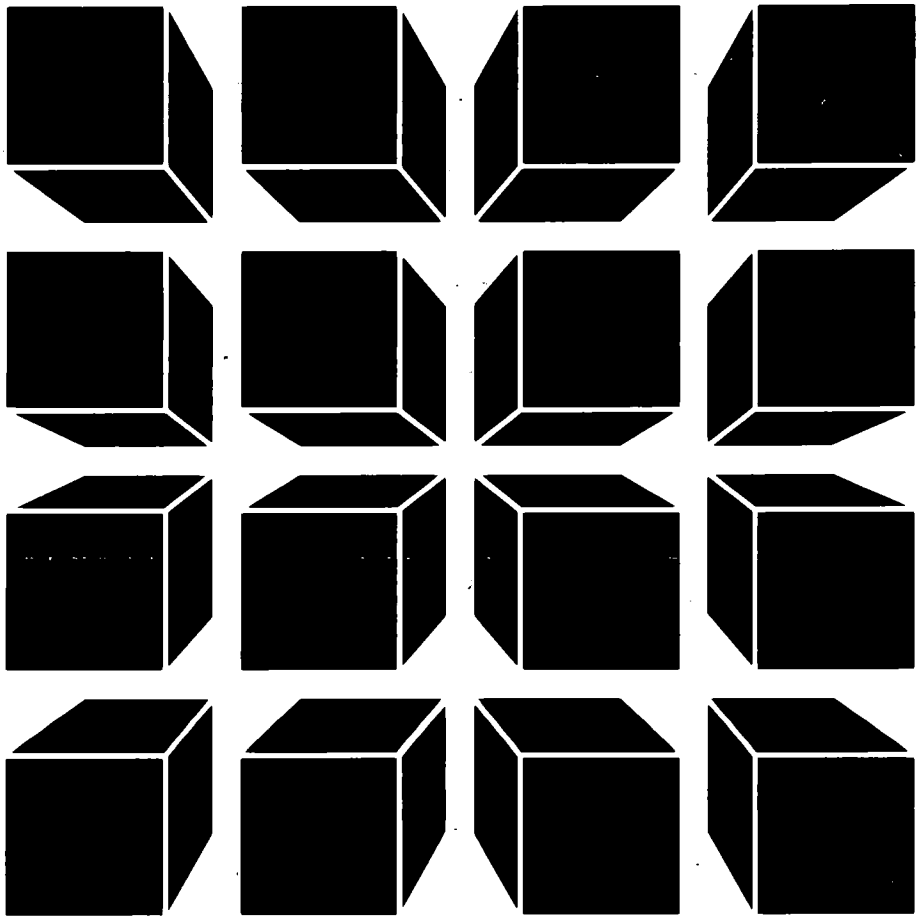
AT&T VIDEOTAPE LIBRARY

FUNDAMENTALS OF THE UNIX® SYSTEM - ADVANCED LEVEL



WORKBOOK

AT&T

CONTENTS

AT&T COMPUTER SYSTEMS EDUCATION PROGRAM INTRODUCTION TO STUDENT WORKBOOK

To the Student,

Welcome to the AT&T Videotape Library. We hope you will enjoy using the videotape lessons and this workbook. You will find the material both practical and easy to follow. You will be using what you learn in your daily work almost immediately.

If you haven't taken a class by videotape before, you are in for an exciting and fun experience, because this medium gives you control over the pace of your learning. Each videotape lesson is divided into segments which are clearly marked with colored panels, so you can find them easily while using fast forward or rewind. When you encounter new terms or concepts, you can immediately review and reinforce them by using the videotape player's rewind feature. If you find some material that is already familiar to you, simply use the player's fast forward button to jump ahead to the next segment. You can even stop the tape completely to take a break. The lesson will pick up right where you left off, and you won't miss a word.

This course, on **Fundamentals of the UNIX® System, Advanced Level**, has been designed for users who have an intermediate level understanding of the UNIX System. You will gain a mastery of UNIX System fundamentals through shell programming tools and procedures. The course also introduces you to the UNIX System DOCUMENTERS WORKBENCH™ text processing software.

This workbook has been prepared as a support tool for this videotape course. Each volume in the workbook follows one lesson. At any point in the lesson, you can stop the tape and use the workbook to review examples or descriptions. This workbook also contains summaries and reviews of what you learned in each videotape lesson, and a job aid referring to new commands or concepts. At the end of each lesson, there are exercises that test your knowledge of the material you have just studied. If you have trouble with any of the exercises, it would be a good idea to go back and review that section of the lesson.

VOLUME 1

Shell Procedures and Background Processing

WHY ARE SHELL PROCEDURES VALUABLE?

- Eliminate retyping frequently-used commands.
- Simplify executing complex commands.
- Allow customizing UNIX System commands.
- Allow creating new commands.

CREATING AND EXECUTING SHELL PROCEDURES

1. Create a file using an editor (**vi**).
2. Make the file executable.
3. Type the file name to execute.

$ **vi nfiles**<CR>

```
# This shell program reports the number
# of files in the current directory

echo  the number of files:
ls | wc -l
```

$ **nfiles**<CR>

nfiles: execute permission denied

$ **chmod 700 nfiles**<CR>

$ **nfiles**<CR>

the number of files:
    6
$

contents of **addx**

```
chmod 700 $*
```

$ addx lookup nfiles nprint1<CR>

contents of **sorter**

```
sort -f -r +1 -2 $*
```

$ sorter phone<CR>

Sue	White	914-798-0021
Robert	Smith	215-645-0792
Marge	Smith	512-453-7564
Bill	Serzo	314-654-4352
sue	phillp	415-436-7934
Bill	Joy	415-768-4236
Rose	Grey	215-567-0798
Joy	Carbone	612-896-0453

QUOTING

What happens if you want to use a special character as itself rather than use its special meaning? Quoting is used to take away the meaning of special characters. There are three types of quoting — quoting using a *backslash*, using *single quotes*, or using *double quotes*.

backslash:

A backslash (\) turns off the special meaning of the next character that you type. A backslash on a command line informs the shell that the character immediately following it is to be taken literally. The backslash is removed from the command line before the command is executed.

single quotes:

All characters between single quotes are taken literally. Once the shell sees the first single quote, it ignores any special characters until it sees the ending single quote. In addition, the characters between single quotes are treated as a single word by the shell. The single quotes are removed from the command line before the command is executed.

QUOTING

- Takes away the meaning of special characters
- 3 types of quoting

\	Character following backslash is literal.
' '	**All** characters between single quotes are literal.
" "	**Most** characters between double quotes are literal. Exceptions: $ \ " `

CONDITIONAL CONSTRUCT-EXAMPLE

contents of **timecard**

```
if
    date | grep  Thu
then
    tput  bel
    tput  bel
    tput  bel
    echo
    echo  ">> Hey $LOGNAME, fill in your TIME CARD!"
    echo
fi
```

Add the **timecard** command to your *.profile*

tput bel rings bell at your terminal

THE TEST COMMAND

- Performs string comparisons
- Performs numeric comparisons
- Determines file attributes

contents of **nprint1**

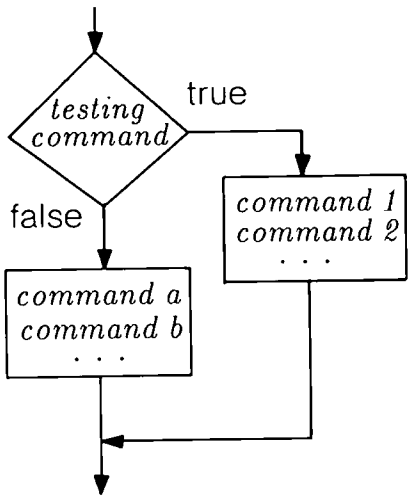
```
echo 'Enter the file you want to print?'
read  name

echo 'Print file with line numbers?'
read  ans
if
  test  "ans"  =  y
then
  pr  -n  -t  $name
fi
if
  test  "$ans"  =  n
then
  cat  $name
fi
```

CONDITIONAL CONSTRUCT

if-then-else

```
if
    testing command
then
    command 1
    command 2
      . . .
else
    command a
    command b
      . . .
fi
```



BACKGROUND EXECUTION

```
$ ps<CR>                                    ←display
  PID TTY          TIME COMMAND             process
  222 tty35        0:04 sh                  information
 5252 tty35        0:01 ps
 3765 tty35        0:10 find

$ kill -9 3765<CR>


$ ps<CR>
  PID TTY          TIME COMMAND
  222 tty35        0:04 sh
 5254 tty35        0:01 ps
```

VOLUME 1 EXERCISE

2. Copy the *lister* file to the file named *clist*. Make the following changes to the *clist* program.

   a. Change the first line to display the words:
   This is a column listing of the current directory:

   b. Change the third line so that a long listing is no longer executed. Instead, the third line should list the files in the current directory in columns (**-C** option). Run the *clist* program.

VOLUME 1 EXERCISE - ANSWERS

1. *lister* program:

```
# This program prints the path name of the current directory.
# It then executes a long listing of the current directory.

echo This is a long listing of the current directory:
pwd
ls -l
```

2. *clist* program:

```
# This program prints the path name of the current directory.
# It then lists the contents of the current directory in columns.

echo This is a column listing of the current directory:
pwd
ls -C
```

3. *printer* program:

```
# This program displays files double-spaced, with line numbers,
# without the header and trailer, and offset by 10 spaces.
# The files to be displayed are typed on the command line.

pr -d -n -t -o10 $*
```

VOLUME 1 SUMMARY

The Shell

SPECIAL CHARACTERS		
Character	Description	Example
\	Backslash. Character following backslash is literal.	$ echo cost is \$5.00<RET> cost is $5.00
' '	Single quotes. All characters between single quotes are literal.	$ echo '* HI *'<RET> * HI *
" "	Double quotes. Most characters between double quotes are literal. Exceptions: $ \ " `	$ echo "* $TERM *"<RET> * 5425 *

CONDITIONAL CONSTRUCTS AND TESTING		
	Description	Format
if-then	Used to test a condition. If testing command is true, the commands between **then** *and* **fi** are executed.	**if** *testing command* **then** *commands* **fi**
if-then-else	Used to test a condition. If testing command is true, the commands between **then** and **else** are executed. If testing command is false, commands between the **else** and **fi** are executed.	**if** *testing command* **then** *commands* **else** *commands* **fi**
test	Performs string comparisons, numeric comparisons, determines file attributes.	test "$TERM" = " " test "$ans" = y

VOLUME 2

Searching, Sorting and Comparing Files

COUNTING LINES, WORDS, AND CHARACTERS

contents of **phone** file

Bill	Serzo	314-654-4352
Rose	Grey	215-567-0798
Marge	Smith	512-453-7564
Bill	Joy	415-768-4236
Robert	Smith	215-645-0792
Sue	White	914-798-0021
Joy	Carbone	612-896-0453
sue	phillip	415-436-7934

```
$ wc  phone<CR>
   8    24    193 phone

$ who ¦ wc  -l<CR>
       11
$
```

```
parts1:C1987653  Wing Nut   Wafer Supplies  .83  360
parts1:C2345643  3/4" nut   RC Electronics  .42  120
parts2:Z2365432  1/4" nut   Global House    .25  620
$
```

The file name *parts** matches file names beginning with *parts*. In this case, there were two files in the current directory: *parts1* and *parts2*.

The **-c** option displays the number of lines containing the pattern. Below, the **-i** and the **-c** options are used together.

```
$ grep -ic nut parts1<CR>
2
$
```

If the pattern contains more than one word, or any special characters, be sure to enclose it in single quotes. Quotes tie multiple words together as a single argument and also protect special characters from interpretation by the shell. Below, **grep** is used in a pipeline to single out lines containing a particular regular expression. A long list of only the subdirectories is displayed:

SORTING LINES

Bill	Serzo	314-654-4352
Rose	Grey	215-567-0798
Marge	Smith	512-453-7564
Bill	Joy	415-768-4236
Robert	Smith	215-645-0792
Sue	White	914-798-0021
Joy	Carbone	612-896-0453
sue	phillip	415-436-7934

```
$ sort  phone<CR>                          ← ASCII
Bill    Joy      415-768-4236                 sort
Bill    Serzo    314-654-4352
Joy     Carbone  612-896-0453
Marge   Smith    512-453-7564
Robert  Smith    215-645-0792
Rose    Grey     215-567-0798
Sue     White    914-798-0021
sue     phillip  415-436-7934
$ sort  -r  phone > revphone<CR>  ← reverse
                                     sort
```

SORTING LINES ON A FIELD

In addition to sorting on the entire line, sorting can be done on particular fields within the lines. If fields are not specified, sorting is done on the entire line. By default, **sort** thinks of a field as including the leading blanks (tabs and spaces) followed by a sequence of non-blank characters. Fields are numbered beginning with 0. The blanks at the beginning of the line are considered to be part of field 0. Field 1 begins at the next sequence of blanks, and so forth. The **-b** option (**-b** stands for **b**lank) is frequently used to ignore the leading blanks in fields, and it begins sorting each field on the first non-blank character.

Suppose the file *books* contains these lines:

```
Wuthering Heights 1818
Huckleberry Finn 1835
Oliver Twist 1812
Treasure Island 1850
Gulliver's Travels 1667
Robinson Crusoe 1660
```

SORTING LINES ON A FIELD

The number following the plus (+), in this case 1, is the field on which to begin sorting. The number following the minus (-), in this case 2, is the field before which to stop the sort. If two entries are identical, an ACSII sort (starting at the beginning of the lines) is done to decide the order.

If the fields of the file are not separated by tabs or spaces, a different field separator can be specified using the **-t** option followed by the appropriate separator. This and other options are discussed on the **sort** manual pages.

EXTRACTING FIELDS

While **grep** singles out lines having a particular pattern, **cut** selects fields (columns) of data. The default field separator (delimiter) is a single <TAB> character. The **-f** option is used to specify the fields to cut. Following the **-f** can be one field, a list of fields (comma separated), or a range of fields (with a hyphen between fields). Note:

1. The **cut** command expects exactly ONE TAB between fields. If a line has two consecutive tabs, this constitutes a "null", or missing field.
2. The **-f** option has a different meaning with **cut** than it has with **sort**. The **-f** option with the **sort** command is used to treat lower case letters like upper case. The **-f** option with the **cut** command specifies the fields to cut.

To cut fields 2 and 3 from the *phone* file, type:

```
$ cut  -f2,3  phone<CR>  ← assumes tab is field
Serzo     314-654-4352          separator
Grey      215-567-0798
Smith     512-453-7564
Joy       415-768-4236
Smith     215-645-0792
White     914-798-0021
Carbone   612-896-0453
phillip   415-436-7934
$
      ↑
     tab
```

EXTRACTING FIELDS

0	1	2
Bill	Serzo	314-654-4352
Rose	Grey	215-567-0798
Marge	Smith	512-453-7564
Bill	Joy	415-768-4236
Robert	Smith	215-645-0792
Sue	White	914-798-0021
Joyi	Carbone	612-896-0453
sue	phillip	415-436-7934

```
$ cut -f 2,3 phone<CR>     ← fields
Serzo       314-654-4352
Grey        215-567-0798
Smith       512-453-7564
Joy         415-768-4236
Smith       215-645-0792
White       914-798-0021
Carbone     612-896-0453
phillip     415-436-7934
$
```

VOLUME 2 EXERCISE

Jeff Abrams	Dublin,OH	614-457-6793	puc04!jta	Jan. 26
Cynthia hill	Whippany,NJ	201-456-3127	uvr05!cvh	March 10
Colby Roberts	Piscataway,NJ	201-346-0945	ptcbe!colby	Feb. 24
Linda Moad	Sunnyvale,CA	408-528-8823	sunny!edm	June 20
john Verity	Princeton,NJ	609-639-1412	pruxb!jgv	July 6
linda Barry	Hopewell,NJ	609-639-1257	fr3b2!lgb	July 29
Diane Carlyle	New York,NY	212-421-3231	be3b5!dbc	Dec. 15
Douglas Vasile	Orlando,FL	305-391-6545	fl3b2!drv	Oct. 22
John Press	Williams,VA	804-325-5324	vax780!ebp	May 5
Donna Howe	Los Angeles,CA	213-453-9812	ca3b20!dth	Nov. 16

Having this file (friends) in mind, answer the following questions:

1. What does this command line produce?

   $wc -l friends

2. What command do we use if we want to sort this file so that the area codes are in increasing order?

VOLUME 2 EXERCISE - ANSWERS

1. Shows the number of lines in friends file; in this case, number of people in the file.

2.

   $sort -t"<tab>" +2 -3 friends >newfriends

3. Sorts file friends first and looks for john regardless of being capital or small letters

4.

   $cut -f1 friends >name1

   $cut -f3 friends >phone1

5.

   $paste name1 phone1 >newphone

VOLUME 3

Introduction to Text Processing

INPUT FILE FOR LETTER

```
.ND "Dec 9, 1935"
.WA "J. P. Sparks" "Director"
144 Holloway Road
Paterson, N.J. 07845
.WE
.IA
Conglomerate Investment Inc.
146 Park Avenue South
Suite 1107
New York, New York 10005
.IE
.LO SJ "New Ideas"
.LO SA "Dear J.Q. Biggie:"
.LT BL
.P
This letter is in reference to the documents sent
to you during the last year.
The last document is dated July 19, 1934.
The first has no date but is titled "The New Phase".
I hope you enjoy the reading.
.P
A list is given below which describes the work
being done by my associates.
.BL 5
.LI
Bulbs:
A new way to light up your life.
Not a plant or animal but a glow in the night.
With this invention, the world will be glowing
by night.
```

LETTER OUTPUT

144 Holloway Road
Paterson, N.J. 07845

Dec 9, 1935

Conglomerate Investment Inc.
146 Park Avenue South
Suite 1107
New York, New York 10005

Dear J.Q. Biggie:

SUBJECT: New Ideas

This letter is in reference to the documents sent to you during the last year. The last document is dated July 19, 1934. The first has no date but is titled The New Phase. I hope you enjoy the reading.

A list is given below which describes the work being done by my associates.

- Bulbs: A new way to light up your life. Not a plant or animal but a glow in the night. With this invention, the world will be glowing by night.
- Switches: These devices are used to turn bulbs on and off. An entire industry will be formed to manufacture these devices. The ground floor is wide open in this area.

	Availability	
	Local Customers	International Customers
Bulbs	Spring 1936	Fall 1938
Switches	Summer 1936	Winter 1949

Cordially yours,

J. P. Sparks
Director

tpo

Copy to
A.B. Caro
D.E. Fisk
G.H. Iman

BUSINESS LETTER MACROS

.IE Marks end of inside address.

— Letter's Opening

.LO SJ *[Topic]*

Prints SUBJECT: **Topic**

.LO SA *[Salutation]*

Prints **Salutation** or To Whom It May Concern.

```
.LO SJ "New Ideas"
.LO SA "Dear J.Q.Biggie:"
```

— Business Letter Type

.LT *[arg]*
.LT BL

Blocked Letter

— Paragraphs

.P 0 (Left block this paragraph - default)

.P 1 (Indent the first line of this paragraph with 5 spaces)

BUSINESS LETTER MACROS

```
.BL 5
.LI
Bulbs:
A new way to light up your life.
Not a plant or animal but a glow in the night.
With the invention, the world will be glowing
by night.
.LI
Switches:
These devices are used to turn the bulbs on and off.
An entire industry will be formed to manufacture
these devices.
The ground floor is wide open in this area.
.LE
```

TABLES

— Tables

.TS	Table start macro
options;	Must end with semicolon
formats.	Must end with dot
data	Table data
.TE	End table

tbl is an nroff/troff preprocessor which processes the data between .TS and .TE.

Options specify the general appearance of the table. Data supplied must be consistent with the options and formats specified.

```
.TS
box center tab(:);
c s s
c c c
c|c|c.
:Local Customer:International Customers
=
Bulbs:Spring 1936:Fall 1938

_
Switches:Summer 1936:Winter 1949
.TE
```

— Formal closing macro

.FC*[closing]*

Prints "Yours very truly" or closing given.

```
.FC "Cordially yours,"
.SG tpo
```

— Signature Block

.SG *[initials][1]*

Prints writer's name [and title] below space for signature. Prints initials above closing notations (letter formatter's name).

— NOTATION

.NS *[arg]*

VOLUME 3 EXERCISE

1. Write a letter and change the date to JULY 15, 1990. Also provide recipient's address, subject line, signature line and send a copy of letter to three individuals.

2. List some UNIX System characteristics using bullet list.

3. What does this file produce :

```
.sp 5
.TS
box center tab(:);
c s s s
c | c | c | c
| | | | | | |.
American Presidents
=
Name:Party:Term:Election-opponent

Franklin D. Roosevelt:Democratic:1933-1945:\
1932-Hoover

:::Thomas

:::_
```

VOLUME 3 EXERCISE - ANSWERS

1. Here is the source to produce the letter

```
.ND "JULY 15,1990"
.WA "W. Kerman"
.WE
.IA "Bob Williams" "Personal Chief"
Summit Research Company
38 River Road
Summit, NJ 07902
.IE
.LO SA "Dear Mr. Williams"
.LT BL
.P
I enjoyed meeting with you last Tuesday.
Here, as I promised then, is a description
of my activities with Business Computer System, Inc.
.P
I started work with Technical Writer staff on April 16.
.FC "Sincerely"
.SG
.NS O
J. Foley
E. B. Jones
A. Craven
.NE
```

VOLUME 3 EXERCISE - ANSWERS

3. Here is the table

American Presidents			
Name	Party	Term	Election-opponents
Franklin D. Roosevelt	Democratic	1933-1945	1932-Hoover Thomas
			1936-Landon
			1940-Willkie
			1944-Dewey
Harry S. Truman	Democratic	1945-1953	1948-Dewey Thurmond Wallace
Dwight D. Eisenhower	Republican	1953-1961	1952-Stevenson
			1956-Stevenson

VOLUME 3 SUMMARY
MM BUSINESS LETTER MACROS

Paragraphs	
P	Paragraph **.P***[{0 1 2}]* *0 = left-justified(default)* *1 = indented* *2 = indented except after .LE .DE, .H*
Lists	
AL	*Automatically-incremented list start* **.AL***[{1 A a I i}][text-indent][1]*
BL	*start a bullet list* **.BL** *[text-indent][1]*
DL	Start a dash list **.DL***[text-indent][1]*
VL	*Start a variable tag list* **.VL** *text-indent [mark-indent][1]*
LI	*list item follows; 1 means that mark* is to be prefixed to the current mark **.LI***[mark][1]*
LE	List end; 1 means to output a blank line after list (default - no blank line) **.LE***[1]*
sp	Space vertical distance N in either direction **.sp** *N*