Faculty of Information Technology

University of Moratuwa

Batch 21 – Level 2 Semester 1

Operating System Lab 06

TITLE: Shell Scripting- Database.

OBJECTIVE: To study database handling using shell script.

Theory:
Essential commands used:

CUT

Like sort, cut is also a filter. True to its name, it cuts or picks up a given number of character or fields from the specified file.

Eg:

```
$echo I like cricket | cut -b 1
```

Say we have a large database of employee information. If from this we want to view only a few selected fields, for instance name and division, cut is answer. If the name happens to be the second field and the division the seventh we would say,

```
$ cut –f 2,7 empinfo
```

If we are to view fields 2 through 7 we can say,

```
$cut –f 2-7 empinfo
```

The cut command assumes that the fields are separated by tab character. If the fields are delimited by some character other than the default tab character, cut supports an option –d which allows us to set delimiter. The file *empinfo* may have the information for each employee stored in the following format.

```
Name: age: address: city: pin: division
```

Each piece of information is separated by colon, hence we require the filed delimiter to be recognized as ':'. The command for listing the name and division fields would now be,

```
$cut –f 2,7 –d":" empinfo
```

The cut command can also cut specified columns from a file and display them on the standard output. The switch used for this purpose is –c. For example,

```
$ cut –c 1-15 empinfo
```

As a result, the first 15 columns from each line in the file empinfo would be displayed.

GREP

grep is an acronym for 'globally search a regular expression and print it'. The command searches the specified input fully (globally) for a match with the supplied pattern and displays it. While forming the patterns to be searched, we can use shell metacharacters, or regular expression, as professional UNIX users call them. Knowing the versatility of the metacharacters,what powers they yield to grep can easily be imagined. Added to that is its capability to search in more than one file, enhanced by the use of various options or switches.

The simplest form of grep command would be:

```
$ grep picture newsfile
```

This would search the word 'picture' in the file *newsfile* and if found, the lines containing it would be displayed on the screen.
To tailor even the weirdest criteria for searching, grep has at his disposal the metacharacters.

```
$ grep [Rr]ebecca myfile
```

Here grep would search for all occurrences of 'Rebecca' as well as 'rebecca' in *myfile* and display the lines which contain one of these words.

```
$ grep b??k myfile
```

This command would display all four letter words whose first letter is a 'b' and last letter, a 'k'. The two '?' symbols represent one character each. Thus, lines book, back, beak, etc. would be listed.

One may wish to see only those lines which do not contain the search patterns, the option –v makes this possible.

```
$grep –v a* myfile
```

Option	Meaning
-i	Ignores case while searching.
-l	Returns only file names containing a match, without quoting the text
-n	Returns line number of matched text, as well as the text itself.
-s	Suppresses error messages
-v	Returns lines that do not match the text.
-c	Returns only the number of matches, without quoting the text.

References:

1. Unix Concepts and Applications By Sumitabha Das, Tata McGraw Hill
2. UNIX Shell Programming by Yashwant Kanetkar, BPB Publications.