UNIX and Shell Programming

Assignment 7: sed and awk

**Exercise 1**

* 1. Launch a terminal.
  2. Create and save the following file. Note that the fields are sepa- rated by one or more spaces (randomly). The first field is the first name, the second field is the last name, and the third field is the age.

John Adams 55

George Bull 77

Anne Blue 99

Janet Blue 67

Ben Benjamin 78

Ted White 32

* 1. Use a one-line sed command to reorganize the file using the comma/tab pattern shown in the following file. Note that the last name is before the first name, and there is only one space between the names and a space before the numbers. Use the same name for the new file.

Adams, John 55

Bull, George 77

Blue, Anne 99

Blue, Janet 67

Benjamin, Ben 78

White, Ted 32

* 1. Sort the file first according to the last name and then according to the age. Watch out for the comma after the last name. Use the same name for the new file.
  2. Use a sed script and a sed command to put a set of five asterisks at the beginning and end of each line that contains the pattern Blue.
  3. Use the cat command to insert a line number at the beginning of each line in a file.
  4. Write a sed script and a sed command to split the file into three files. The first file, called f1, contains lines 2 and 3. The second file, called f2, contains lines 4 and 5. The third file, called f3, contains lines 1 and 6.
  5. Print all of the files created in this session and verify the output.
  6. Quit the terminal.

# Exercise 2

* 1. Launch a terminal.
  2. Create the following file and call it a7-e4-f1. Each line in the file is an absolute pathname of a file.

bin/date bin/programs/cal usr/bin/date usr/report/file1 usr/report/1etters/lett1

/spool/mails

* 1. Write a sed script (a7-e4-f2) and a sed command to extract the lowest level direc- tory and the name of the file from the path (separated by spaces) and store it in a file called a7-e4-f3. The file should look like the following (directory then file}:

/bin date

/bin/programs cal

/usr/bin date

/usr/report file1

/usr/report/letters lett1

/spool mails

* 1. Quit the terminal.

# Exercise 3

* 1. Launch a terminal.
  2. Create the following file and call it a7-e6-f1. The file is a C program that multiplies two numbers. It contains some comments which begin with the two-character token

/\* and end with the two-character token \*/. In this program, comments can be on one line or can span more than a line

/\* This program reads two integer numbers from the keyboard and prints their product.

Written by:

Date:

\*/

/\* Statements \*/

scanf ("%d", &number1); scanf ("%d", &number2); result = number1 \* number2; printf ("%d", result); return 0;

} /\* main \*/

* 1. Write a sed script a7-e6-f2.sed and a sed command to delete the comments from the file. Call the new file a7-e6-f3.
  2. Quit the terminal.

# Exercise 4

1. Create the file named workers.txt with the following

First Name Last Name Rate Hours

George White 18.00 23

Mark Red 18.10 20

Mary Blue 10.89 25

Dan Black 12.00 0

Susan Green 18.00 40

Nora Brown 17.20 46

Bruce Purple 12.20 52

John Gray 11.00 39

Bob Gold 15.00 45

Steve Silver 14.67 25

a. Write awk command to print the first and last name who did not work in the last week

b. Write awk command to print the record of the employee whose rate is $15 or more

c. Write awk command to print the record whose first name is Mary

d. Write awk command to print the record of the employee whose rate between $1 and $18

2. Create a file sales.txt with the following contents

**Month Sales**

January 20

February 30

March 43

February 34

January 12

June 89

May 97

June 60

July 23

August 13

August 45

October 56

October 45

November 34

a. Write awk command to find total sales

b. Write awk script to find the total sales in every month

c. Write awk script to find the months with no sales