Exercise 2:

1. Give the AWK script that displays the word frequency of a text.

Awk Script:

```
{
    for (i = 1; i <= NF; i++) {
        word = $i

        if (word != "") {
            words[word]++
        }
    }
}

END {

    for (word in words) {
        print word, " -->  ",words[word]
    }
}
```

Awk command:

```
salmane@salmane-VirtualBox:~/Desktop$ awk -f script.awk -F ':' /etc/passwd
PulseAudio daemon,,,  -->   1
systemd-oom  -->   1
121  -->   2
list  -->   1
102  -->   2
messagebus  -->   1
34  -->   2
3  -->   2
/run/rpcbind  -->   1
1000  -->   2
29  -->   1
10  -->   2
nm-openvpn  -->   1
rtkit  -->   1
/var/cache/man  -->   1
/dev  -->   1
root  -->   2
115  -->   1
tcpdump  -->   1
news  -->   2
x  -->   51
109  -->   1
8  -->   2
Salmane,,,  -->   1
122  -->   2
```

2. Create an AWK script that show the number of repetition of a specific string in a list of strings then inverse it.

Script:

```
BEGIN {
    target_string = ARGV[1]
    ARGV[1] = ""
}

{

    for (i = 1; i <= NF; i++) {
        string = $i

        if (string == target_string) {
            count++
        }
    }
}


END {
    print "Occurrences : \"" target_string "\" --> " count
}
```

Command: awk -f script2.awk -v target_string="specific string" strings.txt

3. Given a list of telephone numbers of the form 123456789 use sed to rewrite them as (123)456-789.

```
salmane@salmane-VirtualBox:~/Desktop$ echo "987654321" | sed 's/\([0-9]\{3\}\)\([0-9]\{3\}\)\([0-9]\{3\}\)/(\1)\2-\3/'
(987)654-321
```

4. Use sed to select and convert all file names with suffix .html given as output by ls into capital letters with suffix .HTM . Check out command y in sed man page

```
salmane@salmane-VirtualBox:~/Desktop$ touch salmane.html keiken.html not.htm
salmane@salmane-VirtualBox:~/Desktop$ ls |  sed 's/\(.*\)\.html$/\U\1.HTM/'
KEIKEN.HTM
not.htm
SALMANE.HTM
script2.awk
script.awk
```

Use sed to extract full user names from /etc/passwd

```
salmane@salmane-VirtualBox:~/Desktop$ sed 's/:.*//' /etc/passwd
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
systemd-network
systemd-resolve
messagebus
```

5. Write a program with the following behavior:

- Threads are created (their number being passed as a parameter when launching the program)
- Each thread displays a message (for exampleHello !)
- The principal thread waits for the termination of the various threads created
- Each thread shows his PID
- Edit the program that each thread receives it's priority & print it.

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <unistd.h>

void* thread_function(void* arg) {
    printf("Hello! My PID is %d\n", getpid());
    return NULL;
}

int main(int argc, char* argv[]) {

    if (argc != 2) {
        fprintf(stderr, "Usage: %s <number_of_threads>\n", argv[0]);
        return 1;
    }

    int num_threads = atoi(argv[1]);
    if (num_threads <= 0) {
        fprintf(stderr, "The number of threads must be a positive integer.\n");
        return 1;
    }

    pthread_t threads[num_threads];


    for (int i = 0; i < num_threads; i++) {
        pthread_create(&threads[i], NULL, thread_function, NULL);
    }

    for (int i = 0; i < num_threads; i++) {
        pthread_join(threads[i], NULL);
    }

    printf("All threads have finished execution.\n");
    return 0;
}

```

```
salmane@salmane-VirtualBox:~/Desktop$ gcc -o threads threads.c -pthread
salmane@salmane-VirtualBox:~/Desktop$ ./threads 5
Hello! My PID is 8557
Hello! My PID is 8557
Hello! My PID is 8557
Hello! My PID is 8557
Hello! My PID is 8557
All threads have finished execution.
```