**Task No. 1:** While utilizing the Linux commands studied so far, provide an example (other than the one shown in this Lab) of a combination of several Linux commands in which pipes are used more than once. Also provide a snapshot of the generated output.

**Solution: **

**Graphical user interface, text

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Task No. 2:** Write C program to print all alphabets from a to z using while loop.(using ASCII).

**Solution:**

#include <stdio.h>

int main() {

char ch=97;

printf("Alphaphet a to z Printing\n");

while(ch<=122){

printf("%s ",&ch);

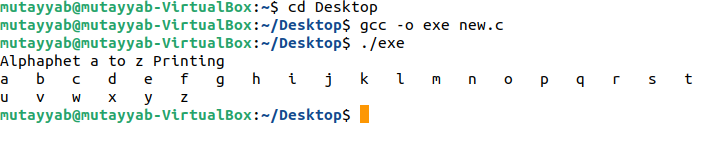
ch++;

}

printf("\n");

return 0;

}

**Output:**

**Task No. 3:** Write a shell script that records the full path of all the files present within a directory into a record.txt file. Along with full path name, the script should also record the number of words, characters and lines within each file.

**Solution:**

#!/bin/bash

cd /

for file in home/mutayyab/Desktop/Important/\*.\*

do

wc -c $file >> output.txt

wc -w $file >> output.txt

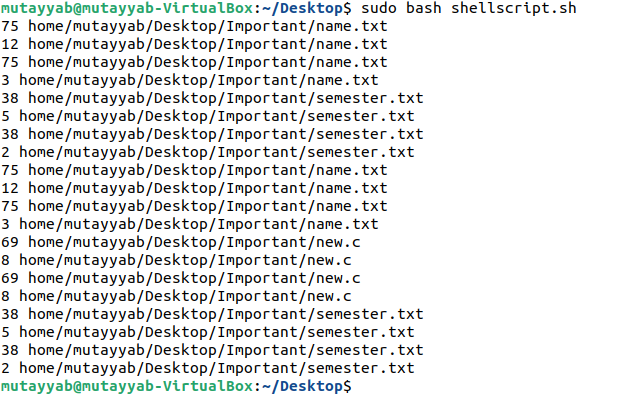
wc -m $file >> output.txt

wc -l $file >> output.txt

done

cat output.txt

**Output:**



**Task No. 4:** Write a C program that asks the user to provide an integer input in the main() function. The program would call a function even\_odd() from the main() function, where the function even\_odd() accepts an integer as an argument, determine and display if the passed integer is either even or odd.

**Solution:**

#include <stdio.h>

void even\_odd(int n);

int main() {

int num;

printf("----------------------------------------------- \n");

printf("-------------Even Odd Calculator------------------ \n");

printf("----------------------------------------------- \n");

printf("\nEnter any integer: ");

scanf("%d", &num);

even\_odd(num);

return 0;

}

void even\_odd(int n) {

if (n % 2 == 0) {

printf("%d is even Number.\n", n);

} else {

printf("%d is odd Number.\n", n);

}

}

**Output:**

