# Procedure:

1. The program will ask user to choose between encryption and decryption (e.g. e for encryption and d for decryption):

**echo** "Choose between Encryption (enter e) and Decryption (enter d): "

**read** ch

**echo**

ask the user to choose between decryption and encryption and store input in ch



2. If the user enters ‘e’:

a. The program should print on the screen “Please input the name of the plain text file”

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo** "Please input the name of the plain text file: "

if the user chooses encryption program asks for plain file



b. The program should remove none alphabet characters

c. Convert all characters to lower case

progress**=$(tr -d '[0-9!-/:-@[-`{-~]' < $file | tr '[A-Z]' '[a-z]')**

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo** "proccesed file content :"

**echo** **$progress**

**echo**

**fi**

delete un-alphabetic chars and converting to lower case letters and store the result in progress

A close up of a screen

Description automatically generated

d. After that, the program must print the sum of word characters frequencies e. After that, the program should print shift value

wordcount**=**1

shiftvalue**=**0

numofwords**=$( echo "$progress" |wc -w )**

#seperate words

**while** **[** **$numofwords** **-**ne 0 **]** **;do**

lettercount**=**1 #reset letter index to 1

word**=$( echo "$progress" |cut -d' ' -f $wordcount )**

letter**=$( echo "$word" |cut -c $lettercount )**

total**=**0

#seperate letters and calculate frequency of the word

**until** **[** **-z** **$letter** **]** **;do**

count**=$( echo "$progress" |tr -cd $letter |wc -c )**

lettercount**=$((** **$lettercount+**1 **))**

total**=$((** **$total+$count** **))**

letter**=$( echo "$word" | cut -c $lettercount )**

**done**

#getting the maximum frequency as shift value

freq**=$((** **$total%**26 **))**

**if** **[** **$freq** **-**gt **$shiftvalue** **]** **;then**

shiftvalue**=$freq**

**fi**

**echo** "the frquency for $word : $freq"

wordcount**=$((** **$wordcount+**1 **))**

numofwords**=$((** **$numofwords-**1 **))**

**done**

**echo** "Shift value is $shiftvalue"

cutting each word and calculating its frequency and storing the max frequency in shiftvalue

A close up of a sign

Description automatically generated

f. Ask user to input the name of the cipher text file

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo**

**echo** "enter the name of the cipher text file: "

**read** out

**if** **[** **!** **-f** **$out** **]** **;then**

**echo** "this file doesn't exist!"

**exit** 1

**fi**

**asking user for the cipher file to store the encrypted data**

****

g. The program will write the generated cipher text on the cipher file

**echo** -n "" **>** **$out** #clears the output file

numofchars**=$( echo "$progress" |wc -c )**

**for** i **in** **$( seq 1 $(($numofchars-1)) )** **;do**

char**=$( echo "$progress" |cut -c $i )**

**if** **[** "$char" **=** " " **]** **;then**

**printf** ' ' **>>** **$out**

**continue** **;fi**

ascii**=$( printf "%d" \'$char )**

v**=$((** **$ascii+$shiftvalue** **))** #ascii value of shifted character

**if** **[** **$v** **-**gt 122 **]** **;then** #if ascii > z then return to a..

v**=$((** **$v-**26 **))**

**fi**

v**=$( printf '%x' $v )**

**echo** **$v** **|xxd** -p -r **>>** **$out** #converts from hexa ascii to character and prints to file

**done**

**echo** "File has been encrypted!!"

**echo**

**echo** "The encrypted content is:"

**cat** **$out**

**echo**

**exit** 0

clear the output file and prints the encrypted data in it then confirming the user that it’s done

A close up of a sign

Description automatically generated



3. If the user enters ‘d’:

a. The program should print on the screen “Please input the name of the cipher text file”

**elif** **[** **$ch** **=** 'd' **]** **;then**

**echo** "Please enter the name of the cipher text file: "

ask user to enter cipher file name

b. After that, the program must print the sum of word characters frequencies

c. After that, the program should print shift value

wordcount**=**1

shiftvalue**=**0

numofwords**=$( echo "$progress" |wc -w )**

#seperate words

**while** **[** **$numofwords** **-**ne 0 **]** **;do**

lettercount**=**1 #reset letter index to 1

word**=$( echo "$progress" |cut -d' ' -f $wordcount )**

letter**=$( echo "$word" |cut -c $lettercount )**

total**=**0

#seperate letters and calculate frequency of the word

**until** **[** **-z** **$letter** **]** **;do**

count**=$( echo "$progress" |tr -cd $letter |wc -c )**

lettercount**=$((** **$lettercount+**1 **))**

total**=$((** **$total+$count** **))**

letter**=$( echo "$word" | cut -c $lettercount )**

**done**

#getting the maximum frequency as shift value

freq**=$((** **$total%**26 **))**

**if** **[** **$freq** **-**gt **$shiftvalue** **]** **;then**

shiftvalue**=$freq**

**fi**

**echo** "the frquency for $word : $freq"

wordcount**=$((** **$wordcount+**1 **))**

numofwords**=$((** **$numofwords-**1 **))**

**done**

**echo** "Shift value is $shiftvalue"

cutting each word and calculating its frequency and storing the max frequency in shiftvalue

A close up of a sign

Description automatically generated

d. Ask user to input the name of the plain text file

**else**

**echo**

**echo** "enter the name of the plain text file: "

**read** out

**if** **[** **!** **-f** **$out** **]** **;then**

**echo** "this file doesn't exist!"

**exit** 1

**fi**

**asking the user for plain text file to print the decrypted result**



e. The program will write the generated plain text on the plain text file

**echo** -n "" **>** **$out** #clears the output file

#do as encryption but with shifting backward

numofchars**=$( echo "$progress" |wc -c )**

**for** i **in** **$( seq 1 $numofchars )** **;do**

char**=$( echo "$progress" |cut -c $i )**

**if** **[** "$char" **=** " " **]** **;then**

**printf** ' ' **>>** **$out**

**continue** **;fi**

ascii**=$( printf "%d" \'$char )**

v**=$((** **$ascii-$shiftvalue** **))**

**if** **[** **$v** **-**lt 97 **]** **;then**

v**=$((** **$v+**26 **))**

**fi**

v**=$( printf '%x' $v )**

**echo** **$v** **|xxd** -p -r **>>** **$out**

**done**

**echo** "File has been decrypted!!"

**echo**

**echo** "The decrypted content is:"

**cat** **$out**

**fi**

**echo**

**exit** 0

clear the output file and prints the decrypted data in it then confirming the user that it’s done

Examples: 1

Encryption:

A screenshot of a cell phone

Description automatically generatedDecryption: A screenshot of a cell phone

Description automatically generated

2:

Encryption:

A screenshot of a cell phone

Description automatically generated

Decryption:

A screenshot of a cell phone

Description automatically generated

Appendex:

#!/bin/sh

#Yousef Ghanem 1172333

#Asaad Halayqa 1172102

#

#ask the user to choose between decryption and encryption

**echo** "Choose between Encryption (enter e) and Decryption (enter d): "

**read** ch

**echo**

#if the user chooses encryption

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo** "Please input the name of the plain text file: "

#if user chooses decryption

**elif** **[** **$ch** **=** 'd' **]** **;then**

**echo** "Please enter the name of the cipher text file: "

#if the user enter invalid value program exits

**else**

**echo** "invalid input!!"

**exit** 1

**fi**

**read** file

**echo**

#check if the file exists

**if** **[** **!** **-f** **$file** **]** **;then**

**echo** "this file doesn't exist!"

**exit** 1

**fi**

**echo** "these are the contents of the file:"

**cat** **$file**

**echo**

#delete un-alphabatic chars and converting to lower case letters

progress**=$(tr -d '[0-9!-/:-@[-`{-~]' < $file | tr '[A-Z]' '[a-z]')**

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo** "proccesed file content :"

**echo** **$progress**

**echo**

**fi**

wordcount**=**1

shiftvalue**=**0

numofwords**=$( echo "$progress" |wc -w )**

#seperate words

**while** **[** **$numofwords** **-**ne 0 **]** **;do**

lettercount**=**1 #reset letter index to 1

word**=$( echo "$progress" |cut -d' ' -f $wordcount )**

letter**=$( echo "$word" |cut -c $lettercount )**

total**=**0

#seperate letters and calculate frequency of the word

**until** **[** **-z** **$letter** **]** **;do**

count**=$( echo "$progress" |tr -cd $letter |wc -c )**

lettercount**=$((** **$lettercount+**1 **))**

total**=$((** **$total+$count** **))**

letter**=$( echo "$word" | cut -c $lettercount )**

**done**

#getting the maximum frequency as shift value

freq**=$((** **$total%**26 **))**

**if** **[** **$freq** **-**gt **$shiftvalue** **]** **;then**

shiftvalue**=$freq**

**fi**

**echo** "the frquency for $word : $freq"

wordcount**=$((** **$wordcount+**1 **))**

numofwords**=$((** **$numofwords-**1 **))**

**done**

**echo** "Shift value is $shiftvalue"

#if the user chose encryption

**if** **[** **$ch** **=** 'e' **]** **;then**

**echo**

**echo** "enter the name of the cipher text file: "

**read** out

**if** **[** **!** **-f** **$out** **]** **;then**

**echo** "this file doesn't exist!"

**exit** 1

**fi**

**echo** -n "" **>** **$out** #clears the output file

numofchars**=$( echo "$progress" |wc -c )**

**for** i **in** **$( seq 1 $(($numofchars-1)) )** **;do**

char**=$( echo "$progress" |cut -c $i )**

**if** **[** "$char" **=** " " **]** **;then**

**printf** ' ' **>>** **$out**

**continue** **;fi**

ascii**=$( printf "%d" \'$char )**

v**=$((** **$ascii+$shiftvalue** **))** #ascii value of shifted character

**if** **[** **$v** **-**gt 122 **]** **;then** #if ascii > z then return to a..

v**=$((** **$v-**26 **))**

**fi**

v**=$( printf '%x' $v )**

**echo** **$v** **|xxd** -p -r **>>** **$out** #converts from hexa ascii to character and prints to file

**done**

**echo** "File has been encrypted!!"

**echo**

**echo** "The encrypted content is:"

**cat** **$out**

**echo**

**exit** 0

#if user chose decryption

**else**

**echo**

**echo** "enter the name of the plain text file: "

**read** out

**if** **[** **!** **-f** **$out** **]** **;then**

**echo** "this file doesn't exist!"

**exit** 1

**fi**

**echo** -n "" **>** **$out** #clears the output file

#do as encryption but with shifting backward

numofchars**=$( echo "$progress" |wc -c )**

**for** i **in** **$( seq 1 $numofchars )** **;do**

char**=$( echo "$progress" |cut -c $i )**

**if** **[** "$char" **=** " " **]** **;then**

**printf** ' ' **>>** **$out**

**continue** **;fi**

ascii**=$( printf "%d" \'$char )**

v**=$((** **$ascii-$shiftvalue** **))**

**if** **[** **$v** **-**lt 97 **]** **;then**

v**=$((** **$v+**26 **))**

**fi**

v**=$( printf '%x' $v )**

**echo** **$v** **|xxd** -p -r **>>** **$out**

**done**

**echo** "File has been decrypted!!"

**echo**

**echo** "The decrypted content is:"

**cat** **$out**

**fi**

**echo**

**exit** 0