Affine Cipher

Mohamed Salem Ali – Faculty of Science -- 3rd year—2nd semester

Problem Description

Affine Cipher project protect your messages from attacker and make attacking process difficult and if the attacker could hacking you it makes your messages inexplicable to unauthorized reader , the program make some steps to convert the messages “plain text” to inexplicable text “cipher text”. (1) get the message “plain text” from program user. (2) get both keys “key a “and “key b” from program user these keys used in “Encrypt Method” and “Decrypt Method”. (3) check the “key a“has an inverse key or not if hasn’t “Encrypt Method” will not make until the user enter key has inverse. (4) Encryption is made, and the output called “cipher text”.

And we can reconvert the “cipher text” to the original message “plain text” the program makes that by some steps: (1) get the cipher text that outputted from “Encrypt method”. (2) get same keys from Encrypt Activity. (3) Decryption is done, and the output is the original message..

Method and Class Description

In this project there are 7 java classes (1) MainActivity (2) HomeActivity (3) Encryption (4) Decryption (5) NotificationClass (6) KeyCeck (7) TextCheck

and 5 activities (“activity_main” , “activity_home” , “activity_encryption” , “activity_decryption”).

---

(1) MainActivity.java

This is the first java class run in the program it includes main method “onCreate”

(1) onCreate this function made connect “activitymain” with this class and create opening activity that be active for couple of seconds and to make splash and switch between two activities and use to switch to homeActivity after splash time is out.

---

(2) HomeActivity.java

This java class run in the program after “activity_main” it includes 7 methods (“on Create”,“decr.setOnClickListener”,“encr.setOnClickListener”,“sendOnEncryption” ,“sendOnDecryption”,“openEncryption”,“openDecryption”)

(1) onCreate this function made connect “activity_home” with this class and to initialize the components (encryption button , decryption button) which in activity.

(2) decr.setOnClickListener this function used to initialize notificationManager decr button and to call sendOnDecryption and openDecryption in onClick methods.

(3) encr.setOnClickListener this function used to initialize notificationManager

encr button and to call sendOnEncryption and openEncryption in onClick method.

(4) sendOnEncryption this function used to show notification after clicking on the encryption button and have small icon and content title “Affine Cipher” and content Text “Encryption Phase”.

(5) sendOnDecryption this function used to show notification after clicking on the decryption button and have small icon and content title “Affine Cipher” and content Text “Decryption Phase”.

(6) openEncryption this function used to change activity to encryption and encryption class and be called in onClick method.

(7) openDecryption this function used to change activity to Decryption and Decryption class and be called in onClick method.

(3) Encryption.java

This java class includes 10 methods (“encryptMessage”,“onCreate”, “encr.setOnClickListener”,“CheckText” “onTextChanged” “check_a” “isStringOnly” “openDialog1” “openDialog2”)

(1) encryptMessage this function take 3 parameters and encrypt msg by using a, b and return ciphertext which equals (a*msg+b)%26.

(2) onCreate this function made connect “activity_encryption” with this class and to initialize the components (encrypt button, a edittext, b edittext, plaintext edittext, ciphertext textview) which in activity.

(3) encr.setOnClickListener this function used to encrypt plain text by calling Encrypt() or openDialog1() in openDialog2() methods.

(4) CheckText this is a TextWatcher function used to make button not active until user write the plaintext and keys.

(5) onTextChanged this function check that a key and b key and plaintext not empty and if they are not empty make the encrypt button active.

(6) check_a here check that the a key that user entered is valid and correct and return true if the key is correct and false if the key is not valid.

(7) isStringOnly this is a boolean function check that the plaintext is english letters only and return true if the plaintext is correct.

(8) openDialog1 this function used to show alarm dialog by call java class KeyCheck().

(9) openDialog2 this function used to show alarm dialog by call java class TextCheck()

(10) Encrypt this function used to call encryptmessage function from class.

(4) Decryption

This java class includes 11 methods (“decryptMessage”, “modInverse”,“onCreate”, “decr.setOnClickListener”,“CheckText” “onTextChanged” “check_a” “isStringOnly” “openDialog1” “openDialog2”)

(1) decryptMessage this function takes 3 parameters and decrypt msg by using a, b and return plaintext.

(2) modInverse this function has two integer parameters return the inverse of key.

(3) onCreate this function made connect “activity_decryption” with this class and to initialize the components (decrypt button, a edittext, b edittext, ciphertext edittext, plaintext textview) which in activity.

(4) decr.setOnClickListener this function used to decrypt cipher text by calling Decrypt() or openDialog1() in openDialog2() methods.

(5) CheckText this is a TextWatcher function used to make button not active until user write the ciphertext and keys.

(6) onTextChanged this function check that a key and b key and ciphertext not empty and if they are not empty make the decrypt button active.

(7) check_a here check that the a key that user entered is valid and correct and return true if the key is correct and false if the key is not valid.

(8) isStringOnly this is a Boolean function check that the ciphertext is English letters only and return true if the ciphertext is correct.

(9) openDialog1 this function used to show alarm dialog by call java class KeyCheck().

(10) openDialog2 this function used to show alarm dialog by call java class TextCheck()

(11) Decrypt this function used to call decryptmessage function from class.

---

(5) NotificationClass.java

This java class includes 2 methods (“onCreate”,“createNotificationChannel”)

(1) onCreate function made connect “activity_encryption”, “activity_decryption” with this class and to call createNotificationChannel function.

(2) createNotificationChannel his function is used two create two channels one for encryption phase and one for decryption phase.

---

(6) KeyCheck.java

This java class includes main method (“onCreateDialog”)

onCreateDialog this function is used to create dialog if key is incorrect with title “Key A Checker” and message “invalid key” and button “Try Again”.

(7) TextCheck.java

This java class includes main method (“onCreateDialog”)

onCreateDialog this function is used to create dialog if text is incorrect with title “Plain Text Checker” and message “Invalid Text” and button “Try Again”.

Testing

When user open the application, the main activity will appear; it is a splash screen

Like in [Figure 1] it includes textview “Affine Cipher” in the middle of the screen It appears for two second then the home activity open like in [Figure 2] it includes

TextView “Affine Cipher” and Encryption Button and Decryption button.

When user click on the Encryption button there is notification appears like in [Figure 5] and open Encryption activity like in [Figure 3] it includes TextView “Encryption” and EditText “A key” and EditText “B key”and EditText “PlainText” and Button “Encrypt”.

When user click on the Decryption button there is notification appears like in [Figure 6] and open Encryption activity like in [Figure 4] it includes TextView

Decryption and EditText “A key”and EditText “B key”and EditText “CipherText” and Button “Encrypt”.

If user use a key that has no inverse there is alert appears to user like [Figure 7] includes the keys that user must use to select one of them.

If user use a text that not completely English letters there is alert appears to user like [Figure 8] tell him that.

Example (1) Encrypt the word “mammy” by using A key “5” and B key “10” the cipher text will be “skssa” like in [Figure 9].

Example (2) Decrypt the word “skssa” by using A key “5” and B key “10” the plain text will be “mammy” like in [Figure 10].

Example (3) Encrypt the word “CryptanalysisCipher” by using A key “25” and B key “30” the cipher text will be “stqfheredqaiasifbgt” like in [Figure 11].

Example (4) Decrypt the word “stqfheredqaiasifbgt” by using A key “25” and B key “30” the plain text will be “CryptanalysisCipher” like in [Figure 12].

Example (5) Encrypt the word “EgyptIsBeautiful” by using A key “7” and B key “212” the cipher text will be “aygplwmdaeklwzkt” like in [Figure 13].

Example (6) Decrypt the word “aygplwmdaeklwzkt” by using A key “7” and B key “212” the plain text will be “EgyptIsBeautiful” like in [Figure 14].

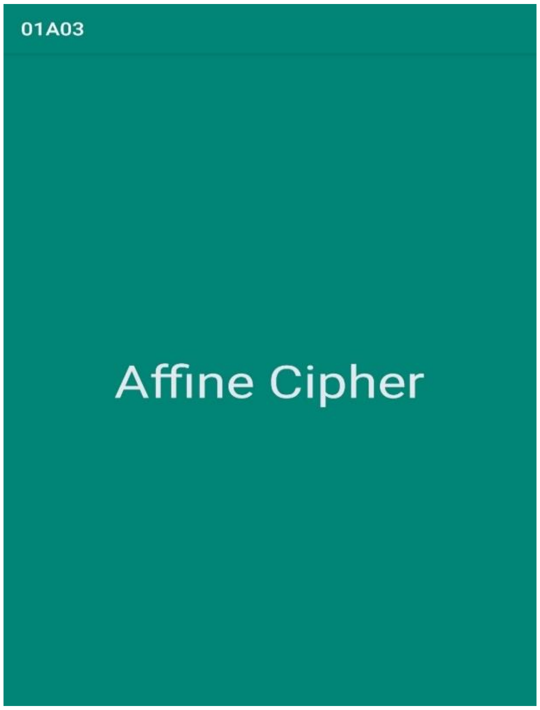


Figure (1).

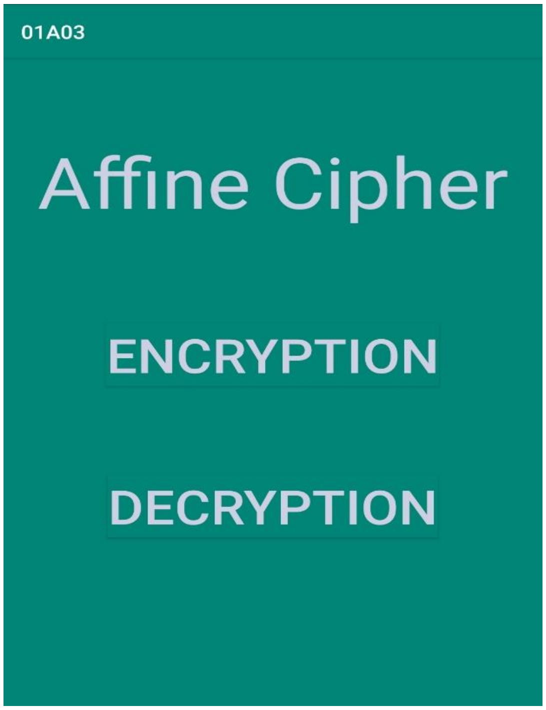


Figure (2).

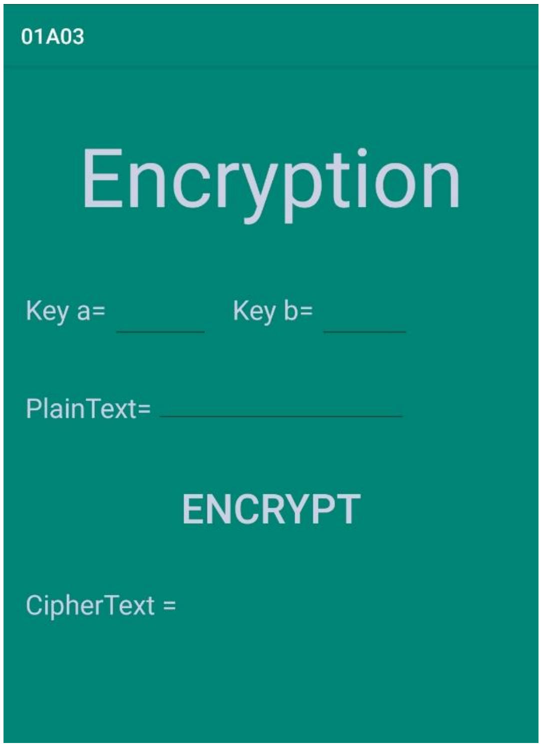


Figure (3).

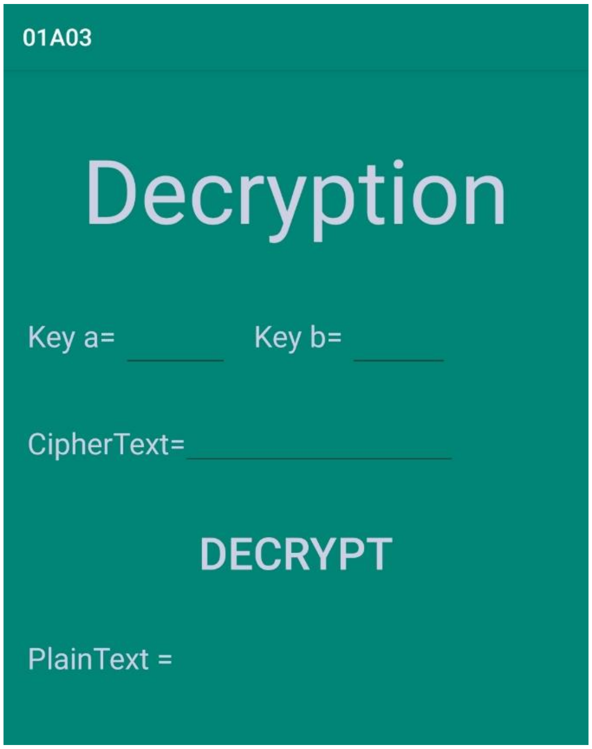
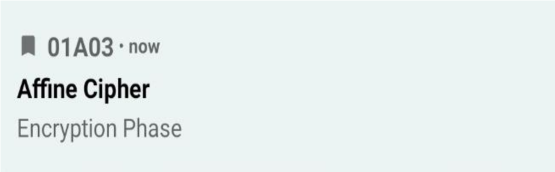


Figure (4).

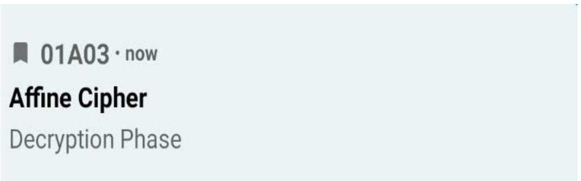


01A03 · now

Affine Cipher

Encryption Phase

Figure (5).

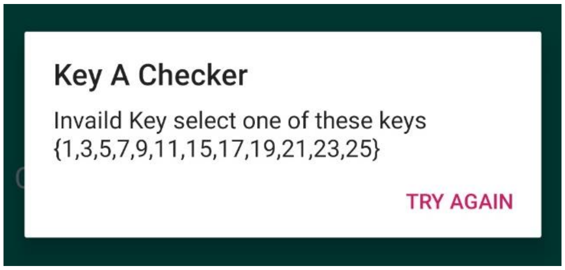


01A03 · now

Affine Cipher

Decryption Phase

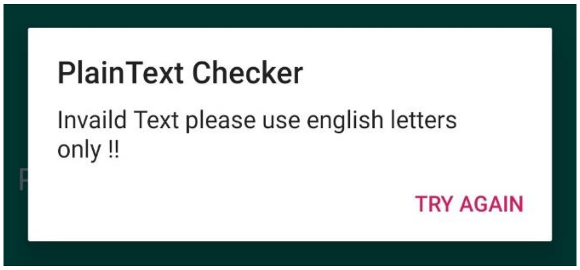
Figure (6).



Key A Checker

Invaild Key select one of these keys {1,3,5,7,9,11,15,17,19,21,23,25}

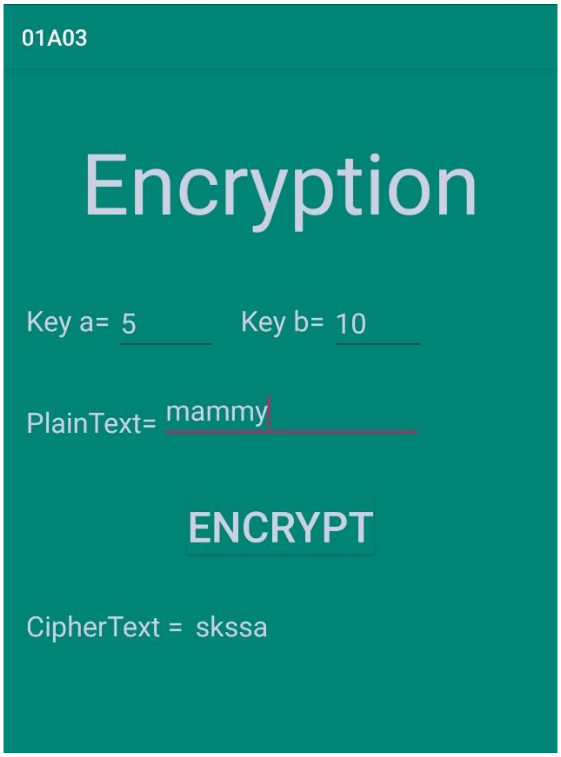
Figure (7).



PlainText Checker

Invaild Text please use english letters only !!

Figure (8).



01A03

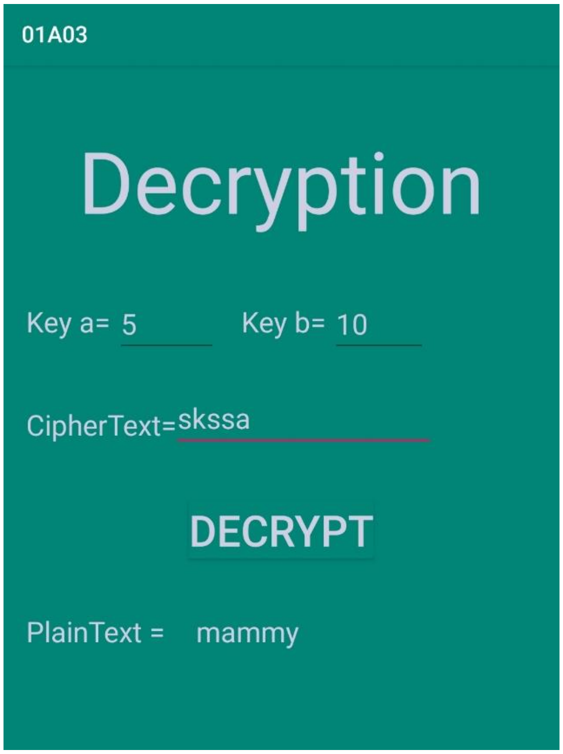
Encryption

Key a= 5 Key b= 10

PlainText= mammy

CipherText = skssa

Figure (9).



01A03

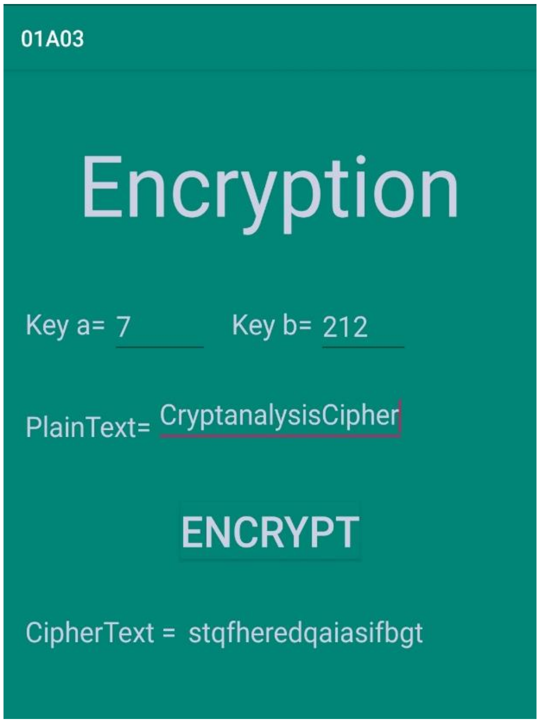
Decryption

Key a= 5 Key b= 10

CipherText=skssa

PlainText = mammy

Figure (10).

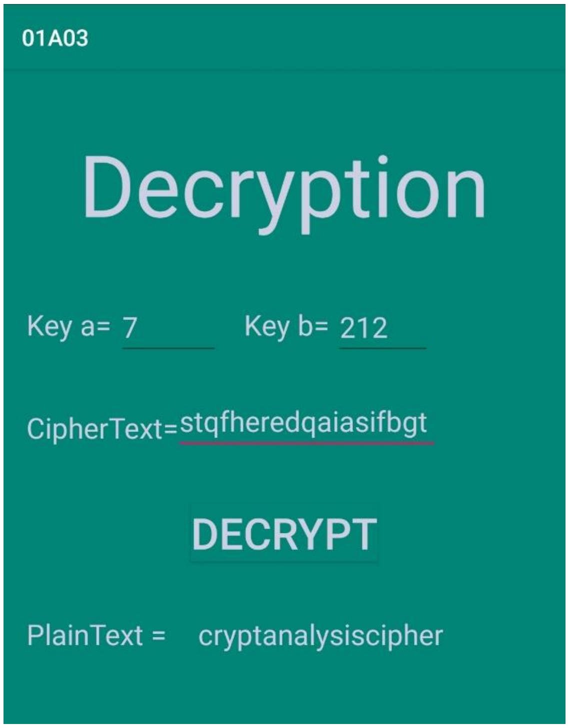


Encryption

Key a= 7 Key b= 212

PlainText= CryptanalysisCipher

CipherText = stqfheredqaiasifbgt



Decryption

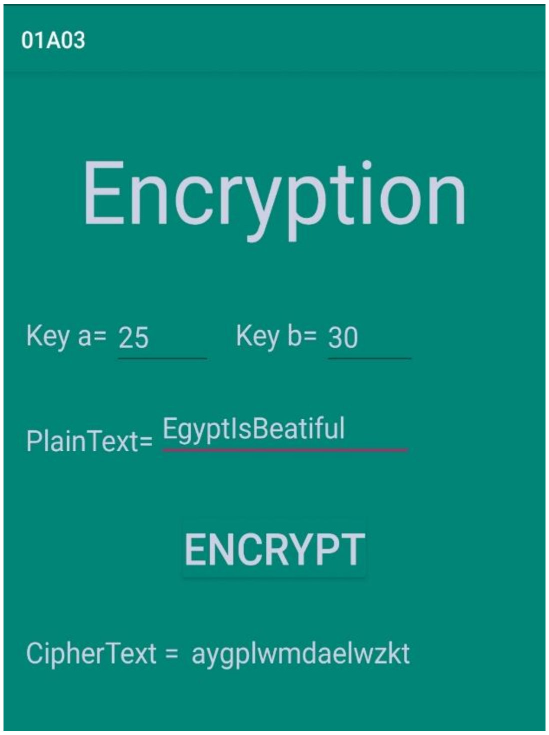
Key a= 7 Key b= 212

CipherText=stqfheredqaiasifbgt

PlainText = cryptanalysiscipher

Figure (11).

Figure (12).

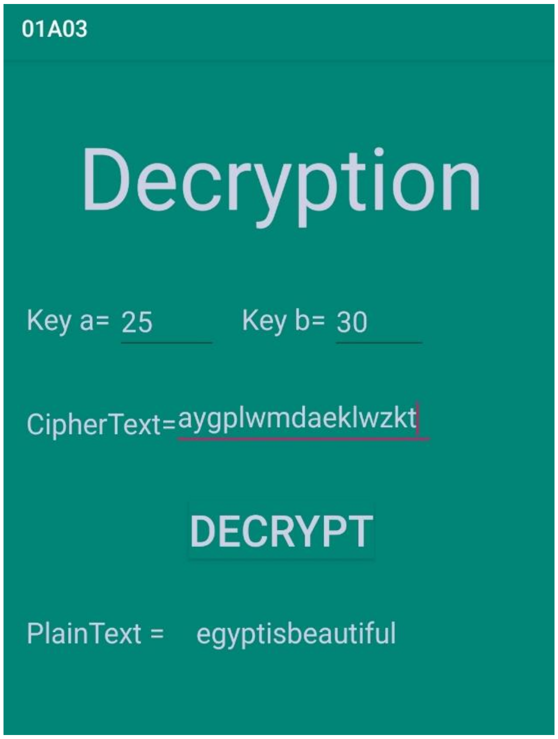


Encryption

Key a= 25 Key b= 30

PlainText= EgyptIsBeatiful

CipherText = aygplwmdaelwzkt



Decryption

Key a= 25 Key b= 30

CipherText=aygplwmdaeklwzkt

PlainText = egyptisbeautiful

Figure (13).

Figure (14).