

Assignment 4 – Encryption Systems



Maximum Points = 50

The purpose of this lab is to focus on the study of classes, objects, GUI, and error checking through exception handling. This is also an opportunity to participate in a pair-programming experience.

“**Encryption** is the process of transforming [information](#) (referred to as [plaintext](#)) using an algorithm (called [cipher](#)) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a [key](#). The result of the process is **encrypted** information (in cryptography, referred to as ciphertext). In many contexts, the word **encryption** also implicitly refers to the reverse process, **decryption** (e.g. “[software for encryption](#)” can typically also perform decryption), to make the encrypted information readable again (i.e. to make it unencrypted).” [<http://en.wikipedia.org/wiki/Encryption>]
Cryptography is the practice of hiding and retrieving information by enciphering and deciphering messages by means of transformations. Cryptanalysis is the process of converting ciphertext into plaintext without knowing the key used in encryption. Cryptology is the study of cryptography and cryptanalysis.

As a cryptographer, you have been asked by the *National Signal Academy (NSA)* to develop a program that includes a collection of encryption techniques. Your program will allow the user to select an encryption technique and encrypt and decrypt messages.

Encryption techniques have been around for thousands of years [<http://csc.colstate.edu/summers/Research/cipher-machines2.ppt>]. One of the oldest and simplest techniques is the Caesar Cipher where each letter of the plaintext is shifted a fixed distance from the original position in the alphabet. For example, if the plaintext is “Java Rocks” and the shift distance is 5, then the ciphertext is “Ofaf Wthpx”. To decipher, you just shift by five in the other direction. You can simplify the work by using two disks similar to the ones shown in Fig. 1.

	<p>A variation of this was developed by Leon Battista Alberti where each letter is replaced by a predetermined letter using a cipher disk (Fig. 1). To decipher, you must have the complete list of pairings (or a cipher disk as shown) and the alignment of the letters.</p> <p>Thomas Jefferson extends this idea to a cylinder with 24 sets of letters. To decipher would require having a cylinder and knowing which ring to use.</p>	 <p>Copyright © Thomas Jefferson Foundation, Inc.</p> <p>Fig. 2</p>
-------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------

Other types of early encryption include:

- 1) The Mexican Army Cipher Disk (1913) with an outer ring of letters and four inner concentric rings of numbers.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	1	2	2	22	2	2	2	2
2	2	2	3	3	3	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	48	4	5	5	5
7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7		9	0	1	2
5	5	5	5	5	5	5	6	6	6	6	6	6	6	6	6	6	7	7	7	7	74	7	7	7	7
3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3		5	6	7	8
7	8	8	8	8	8	8	8	8	8	8	9	9	9	9	9	9	9	9	9	9	10				
9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0				

The users agree before how to rotate the inner four rings and either which ring to use or whether to increment the ring if a letter is repeated. For example “Java Rocks” with 1, 27, 53, and 79 positioned under A, can be –

10 01 22 01 18 14 03 11 19 or
 10 01 22 **27** 18 14 03 11 19, or
 10 27 74 79 18 41 55 89 19, etc.

- 2) The Confederate Cipher Disk, which was designed after the “Vigenère system” which uses a key phrase and a disk with two concentric rings. Position each letter of the message under the letter ‘A’ in the outer ring. Find the letter in the key phrase in the respective position on the outer ring; the cipher letter is below in the inner ring. For example, if the plaintext is “Java Rocks” and the key phrase is “Computer Science”, then the cipher text is “Lohp Lhgbk” To decipher, you must know the key phrase and work backwards.

Your program will read (and display) the contents of a user-selected file [plaintext], apply an encryption algorithm to the clear text to create the ciphertext, and then display (and write) the ciphertext to a user-selected file. Your program should also be able to read (and display) the contents of a file containing the ciphertext and then decrypt contents back into plaintext. Your program will allow the user to select one of at least two different encryption algorithms. For additional challenges, add more encryption algorithms.

Your program should use a GUI with a title and a graphic. The GUI should display both the clear text and the cipher text, provide for a key if used by the cipher, buttons that determine whether the user wants to encrypt or decrypt the message, and a way to select the encryption algorithm to be used.

Plaintext files should all have extensions “txt” and ciphertext files should have extensions “cip”. Create an exception class that catches and handles the exception thrown when the user selects the wrong type file. Handle the exception by displaying an appropriate message and then continue processing.

You must also keep track of the hours you spent on this project and include it with both assignments:

Student name	Requirements Anaysis	Design	Implementation (coding)	Testing

Due before class on Thursday, March 26, 2009) Submit a .doc file containing the UML class diagram showing inheritance for all the classes used in your program and your timesheet. [10 pts]

(Due before class on Thursday, April 2, 2009) Submit your .java files containing your program to the dropbox in WebCT. [50 pts]

Grades are determined using the following scale:

- Runs correctly.....:___/10
- Correct output.....:___/10
- Design of output.....:___/8
- Design of logic.....:___/10
- Standards.....:___/7
- Documentation.....:___/5

[Grading Rubric](#) ([Word document](#))