I’m planning on doing the affine cipher.

My assumptions will be that Java won’t mess up and do something weird with my numbers (like flip over and start doing negative numbers), and that % will work for modular arithmetic.

Methods:

1. Introduction – prints out introductory statement
2. Prompt - Prompts user for phrase to be encoded
3. Encoder/decoder – encodes phrase, prints encoded phrase, then decodes encrypted phrase and prints decoded phrase
4. Error – if the message encoded cannot be encoded, this will print an error message.

Introduction:

Input: nothing

Output: This program will encode and then decode a string using the ROT13 and Affine Ciphers. An Affine Cipher encodes character with an equation (ax + b) mod 26, where x is the ASCII value of the character. ROT13 is a special case of the Affine Cipher, where a is 1 and b is 13.

Prompt:

Input: 5, 8, Hello World!

Output: nothing

Encoder/decoder:

Input: nothing

Output:

Please enter an odd numeric value for a (1-25, not 13): 5

Please enter a numeric value for b: 8

Enter the string you'd like to convert: Hello World!

The Affine encoded string is: Rclla Oaplx!

The Affine conversion was tested and was correct, here's your output: Hello World! The ROT13 encoded string is: Uryyb Jbeyq!

The ROT13 conversion was tested and was correct, here's your output: Hello World!

Error:

Input: numbers that don’t work

Output:

The value for ‘b’ must be between 0 and 26.

Or

The value for ‘a’ must be an odd number between 1 and 26 not including 13.