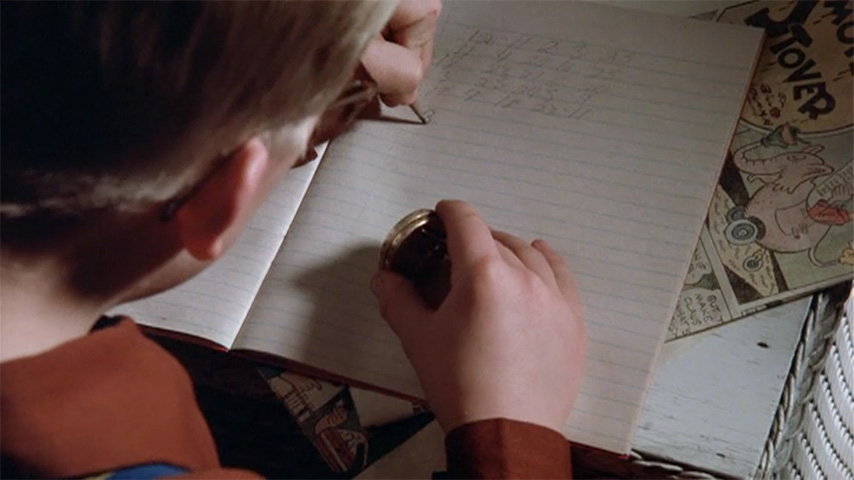
**Lab 10: Secret Codes!**

1. **Background Information**

Since the beginnings of written and verbal communication, people have tried to find ways to convey messages secretly. This might be placing specific keywords in a message that has meanings to a group of people, replacing letters by numbers or other symbols, or coming up with a completely new language that is only understood by certain people. In this lab, you will be trying your hand at encrypting and decrypting messages, using a relatively simple scheme.



**Figure 1: Ralphie and his Little Orphan Annie Secret Decoder Pin from   
“A Christmas Story”**

The scheme we will be using is to offset the ASCII value of the characters in the message using an encryption key. If you remember, a string is stored in MATLAB as a vector of integer numbers we call ASCII values. Each ASCII value corresponds to a specific character that can be displayed by a computer. Below are the ASCII values for the upper and lower case letters.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** |
| 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| **N** | **O** | **P** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** | **Z** |
| 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 |

**Table 1: Capital Letter ASCII Values**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **a** | **b** | **c** | **d** | **e** | **f** | **g** | **h** | **i** | **j** | **k** | **l** | **m** |
| 97 | 98 | 99 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| **n** | **o** | **p** | **q** | **r** | **s** | **t** | **u** | **v** | **w** | **x** | **y** | **z** |
| 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 |

**Table 2: Lowercase Letter ASCII Values**

Assume that we are trying to encode the message: *I have a secret!* If we convert this message into its ASCII values, we would have the following vector:

[73 32 104 97 118 101 32 97 32 115 101 99 114 101 116 33]

To encode the message, we pick an integer value and add this value to the ASCII values in the message. If we pick our key value as 4, the message we would send now becomes:

[77 32 108 101 122 105 32 101 32 119 105 103 118 105 120 33]

If this is displayed as a string, the encoded message becomes: M lezi e wigvix!

To decode the message, we simply perform the opposite: subtract off the key from the ASCII values of the encoded message to recover the original message.

1. **Encrypting a Message**

Write a script that will encrypt a message by implementing the following steps:

1. Prompt the user to enter the message to encrypt and the encryption key that will be used to encrypt the message
2. Convert your message into the ASCII values by changing it from a string into a vector of double values using the double command:

number\_message = double(original\_message);

1. Go through each value in the message and do the following:
   1. If the character is a capital letter (ASCII values 65-90), encrypt it by adding the encryption key value to the ASCII value of the letter, ensuring that the encrypted value remains a capital letter by wrapping around to the beginning of the alphabet (i.e. a Z (ASCII value of 90) would end up as a D (ASCII value of 68) using a key value of 4)
   2. If the character is a lowercase letter (ASCII values 97-122), encrypt it by adding the encryption key value to the ASCII value of the letter, ensuring that the encrypted value remains a lowercase letter by wrapping around to the beginning of the alphabet (i.e. a z (ASCII value of 122) would end up as a f (ASCII value of 102) using a key value of 6)
   3. If the character is not a letter (number, punctuation, space, etc.), leave it as is
2. Display the encrypted message using an fprintf statement and the %s placeholder

Once you have your script written, test it with the following messages and keys:

|  |  |  |
| --- | --- | --- |
| **Message** | **Key** | **Encrypted Message** |
| MATLAB is awesome! | 6 | SGZRGH oy gckyusk! |
| I wanna MATLAB all night, and MATLAB every day! | 18 | A osffs ESLDST sdd fayzl, sfv ESLDST wnwjq vsq! |
| These messages are pretty corny. | 23 | Qebpb jbppxdbp xob mobqqv zlokv. |

**Paste your script below:**

%Models Lab 10 Part B

clear; clc;

original\_message = input('Message to encrypt: ','s');

encryption\_key = input('Encryption key: ');

number\_message = double(original\_message);

for k = 1:length(number\_message)

if number\_message(k) >= 65 && number\_message(k) <= 90

number\_message(k) = number\_message(k) + encryption\_key;

while number\_message(k) > 90

number\_message(k) = number\_message(k) - 26;

end

end

if number\_message(k) >= 97 && number\_message(k) <= 122

number\_message(k) = number\_message(k) + encryption\_key;

while number\_message(k) > 122

number\_message(k) = number\_message(k) - 26;

end

end

end

fprintf('%s\n',number\_message);

1. **Decrypting a Message**

Write a script that will decrypt a message by implementing the following steps:

1. Prompt the user to enter the message to decrypt and the encryption key that will be used to decrypt the message
2. Convert your message into the ASCII values by changing it from a string into a vector of double values using the double command:

number\_message = double(original\_message);

1. Go through each character in the message and do the following:
   1. If the character is a capital letter (ASCII values 65-90), decrypt it by subtracting the encryption key value from the ASCII value of the letter, ensuring that the decrypted value remains a capital letter by wrapping around to the end of the alphabet (i.e. an A (ASCII value of 65) would end up as a W (ASCII value of 87) using a key value of 4)
   2. If the character is a lowercase letter (ASCII values 97-122), decrypt it by subtracting the encryption key value from the ASCII value of the letter, ensuring that the decrypted value remains a lowercase letter by wrapping around to the end of the alphabet (i.e. an a (ASCII value of 97) would end up as a u (ASCII value of 1117) using a key value of 6)
   3. If the character is not a letter (number, punctuation, space, etc.), leave it as is
2. Display the decrypted message using an fprintf statement and the %s placeholder

Once you have your script written, test it by decrypting the messages give below:

|  |  |  |
| --- | --- | --- |
| **Message** | **Key** | **Decrypted Message** |
| Xwln hxd qjen nurvrwjcnm cqn rvyxbbrkun, fqjcnena anvjrwb, qxfnena rvyaxkjkun, vdbc kn j adwcrvn naaxa. | 9 | Once you have eliminated the impossible, whatever remains, however improbable, must be a runtime error. |
| Z nrek kf cvrie kyv nrpj fw DRKCRS reu svtfdv r tfuvi, czbv dp wrkyvi. | 17 | I want to learn the ways of MATLAB and become a coder, like my father. |
| Ocdif gzao viy ocdif mdbco viy ocdif gjr viy ocdif cdbc. Jc, ocz ocdifn tjp xvi ocdif pk da jigt tjp omt! ~Ym. Nzpnn | 21 | Think left and think right and think low and think high. Oh, the thinks you can think up if only you try! ~Dr. Seuss |

**Paste your script below:**

%Models Lab 10 Part C

clear; clc;

original\_message = input('Message to decrypt: ','s');

encryption\_key = input('Encryption key: ');

number\_message = double(original\_message);

for k = 1:length(number\_message)

if number\_message(k) >= 65 && number\_message(k) <= 90

number\_message(k) = number\_message(k) - encryption\_key;

while number\_message(k) < 65

number\_message(k) = number\_message(k) + 26;

end

end

if number\_message(k) >= 97 && number\_message(k) <= 122

number\_message(k) = number\_message(k) - encryption\_key;

while number\_message(k) < 97

number\_message(k) = number\_message(k) + 26;

end

end

end

fprintf('%s\n',number\_message);

1. **Decrypting an Intercepted Message**

You have now been hired by the Engineering Models Acquisition of Information League (EMAIL), which is a super-secret organization tasked with intercepting and decrypting messages. Your current task is to create a script that will attempt to decrypt a message when you do not know the key by doing the following:

1. Prompt the user to enter the message to decrypt
2. Repeat your decryption code from Part C for 25 times, stopping once you’ve correctly decrypted the message  
   ***HINT: create a value for the key and change it each time through the loop, stopping the loop once the message you display makes sense***

To test your script, download the ***Message\_Interceptor.p*** file from Blackboard. When run (you’ll need to type the name at the command window to run it), this code will gather an encrypted message and display it in the command window. After receiving a message, run your script to attempt to decrypt the message. Run this pair three times and record the results below:

|  |  |  |
| --- | --- | --- |
| **Encrypted Message** | **Key** | **Decrypted Message** |
| Mci rcb'h ibrsfghobr! W qcizro vor qzogg. W qcizro pssb o qcbhsbrsf. W qcizr'js pssb gcaspcrm, wbghsor ct o pia, kvwqv wg kvoh W oa. | 11 | You don't understand! I coulda had class. I coulda been a contender. I could've been somebody, instead of a bum, which is what I am. |
| Xiz epo'u zpv dpnf vq tpnfujnf boe tff nf? | 1 | Why don't you come up sometime and see me? |
| Porusg? Ks owb'h uch bc porusg! Ks rcb'h bssr bc porusg! W rcb'h vojs hc gvck mci obm ghwbywbu porusg! | 11 | Badges? We ain't got no badges! We don't need no badges! I don't have to show you any stinking badges! |

**Paste your script below:**

%Models Lab 10 Part D

clear; clc;

original\_message = input('Message to decrypt: ','s');

encryption\_key = 0;

number\_message = double(original\_message);

for n = 1:25

clc;

encryption\_key = n;

for k = 1:length(number\_message)

if number\_message(k) >= 65 && number\_message(k) <= 90

number\_message(k) = number\_message(k) - encryption\_key;

while number\_message(k) < 65

number\_message(k) = number\_message(k) + 26;

end

end

if number\_message(k) >= 97 && number\_message(k) <= 122

number\_message(k) = number\_message(k) - encryption\_key;

while number\_message(k) < 97

number\_message(k) = number\_message(k) + 26;

end

end

end

fprintf('%s\n',number\_message);

correct\_decryption = menu('Is this message correct?','Yes','No');

if correct\_decryption == 1

break

end

end

fprintf('The encryption key is %i\n',encryption\_key);

1. **To be turned in:**

* You will need to upload this word document with all tables, questions, and figures included and the m-file for your script.