Coursework Report

Sonas MacRae
40277542@napier.ac.uk
Edinburgh Napier University - Web Technologies (SET008101)

1 Introduction

Ciphers have been protecting secrets for thousands of years. The invention of mechanical machines (such as the Enigma rotor machine) provided an efficiency to encryption that had never been seen before, this was quickly surpassed thanks to the invention of computers which resulted in encryption techniques too complex for simple pen and paper techniques used in the past.

This web technologies assignment is a combination of writing ciphers in JavaScript and designing a website using HTML and CSS which will be able to run the ciphers. Four ciphers were chosen to be implemented into this website: A Caesar cipher, Number substitution cipher, Fractioned Morse cipher and a Vigenere cipher. The Caesar cipher was chosen because it was easy to implement, and further ciphers could be based off the algorithms used. The Vigenere cipher was based off the Caesar cipher, a similar method was used to encrypt and decrypt. Inspiration to write this cipher came from this website. The number substitution cipher is a simple cipher that substitutes every letter with one of four double-digit numbers. Finally, the Fractioned Morse cipher was implemented, this cipher is more complex than the other three ciphers, it was chosen because it was more challenging to code, the methods used to encrypt and decrypt the Fractioned Morse cipher is different from the other ciphers on the website too. Ciphers were researched to aid this coursework, these are examples of pages used: research website 1, research website 2

2 Design

The design of the website will be easy to use and functional, each cipher will have its own HTML page. The home page will be simple, it will contain a header and four buttons which will be designed to stand out, each button leading to a different page which will have its own cipher. In terms on navigation, the home page will be connected to all of the cipher pages, but none of the cipher pages will be directly linked, the user will have to return to the home page before they can navigate to another cipher page. As a result, this minimizes the number of buttons throughout the website.

Each cipher page will require an input box, output box, buttons for encrypting, decrypting and any other features that will be implemented. Each cipher page will have a "copy output" button which will copy the text from the output box as well as a button which will return the user back to the home page. The buttons will be satisfying to press, they will grow when the mouse cursor hovers over them. To add to this, each page will include a description of the cipher, for example the Caesar cipher page will explain how the cipher works as well as instructions of how to use the cipher. Each cipher page will include links to further reading, embedded YouTube videos which will either be about the cipher itself, or about ciphers in general. The content on the cipher pages will be split up among four boxes in the form of div's, this will make the information easier to digest as the content won't be congested. Every page will host images, the images will not only fill up some of the empty space on the page, they will also make the page more eye-catching, as a result of this the user will focus more of their attention on the page. On the Caesar cipher page there will be a drop-down menu which will hold all of the possible keys for encryption. The Vigenere cipher page and the Fractioned Morse cipher page will have to hold another input box specifically for the key. The Number cipher page won't have any extra elements as it doesn't require a key to encrypt or decrypt. The text font will be carefully selected to compliment the choice of colours, the colour scheme will look professional. All headers on every page will be bigger and bolder than the text in the paragraphs, the reason being the headers have to stand out, this is what the user has to read first, the colour of the text will either be black or white.

Diagram 1 shows how the user can navigate between the different pages, the links are in the form of an arrow pointing to the pages that are accessible from that page. Diagram 2 displays the design of the cipher pages, it shows how the boxes fill up most of the space on the page, and finally diagram 3 shows the design of the index page, it shows the four buttons placed below the title, it's intentionally simple in the hopes that it will make the buttons stand out.

2.1 JavaScript Design

In terms of designing the JavasScript, pseudocode was written to test the difficulty of implementing the ciphers, when researching ciphers, quick solutions were written up before any code was written. Diagram 7 illustrates a solution which can be applied when programming the ciphers, the pseudocode finds the index of any letter in the English alphabet by iterating the letter through an array which stores the alphabet and compares the letter to the current index of the alphabet array and returns the index. This will be essential for the Caesar cipher where the index of the key and the index of the characters in the input will be required to process the encryption. For every cipher, there will be a file containing the JavaScript code, the code will be portioned into functions to reduce repeated code, the functions will be called from within two main functions (encrypt and decrypt), these functions will be called when the corresponding buttons are pressed on the web page. The Caesar cipher and Vigenere cipher will take into account white space when encrypting and decrypting, whereas the number cipher and the fractioned Morse cipher will ignore white space, resulting in the encrypted message having no spaces. The reason being that if the number cipher had spaces after encryption, every word would have an even number of characters, this is too predictable. To erase the spaces, a function will be written which will format the input, this function will replace all spaces with null characters, as well as changing the string to lowercase.

3 Implementation

The navigation of the website works just as planned, the layout of all the pages are as designed, although the final product is a slight improvement upon the design. For example, on the index page each button has an image and has the name of the cipher underneath, all of the buttons do their jobs and the website is easy to use. The design for all of the titles are fancy and easy to read, the buttons stand out and the pages aren't cluttered with unnecessary content. The colour scheme was kept intentionally simple, Bright colours were avoided, yet the content stands out from the background. When the cursor hovers over any of the buttons on any of the pages, they grow, when the cursor is then taken off the button, they go back to their original state. After implementing the design for the box which held the input and output boxes, the layout was slightly changed to make better use of the space, this is shown in diagrams 5.1/5.2.

Diagrams 5.1/5.2 show what the Fractioned Morse Cipher page looks like, the other three cipher pages use the same CSS file so they have the same design, the same colours for the same elements and the boxes are laid out the same, the difference between the cipher pages is the JavaScript attached to them, the text, YouTube videos and the fact that the Number cipher page does not have a key-input.

Diagram 4 shows off the index page, the buttons have now got images on them, they pulsate when the mouse cursor hovers over them, although the buttons are restricted to only the images, clicking the text beneath does nothing.

4 Critical evaluation

This coursework meets all of the requirements of the specification, there is an index page which navigates to ciphers, the pages were designed and implemented. CSS, HTML and JavaScript were used, and the code was stored in appropriate files. The design.html page shows all of the designs used throughout the website. Each page can encrypt and decrypt messages and the cipher pages and has appropriately placed input and output boxes, the same input box is used for encrypting and decrypting which not only saves space but is easier to use. All of the elements on the pages are neatly laid out because a grid system was used to do so, this made adding elements to a page quick and easy.

After comparing this finished coursework to the specification, a handful of possible improvements were found as well as possible improvements to the code. In terms of difficulty, one of the ciphers were hard to implement: the fractioned Morse cipher since it's the only cipher on the website that doesn't swap characters directly. The fractioned Morse cipher is a lot more complex than the other three ciphers, The Vigenere cipher and the Caesar cipher weren't complicated to implement. An improvement to the website would be to include more complex ciphers such as the hill cipher or the Playfair cipher.

Another possible improvement is the buttons on the index page, the images look good, but they're different from one another, a common art style amongst them would be more ideal. Also, the image for the fractioned Morse cipher page button doesn't quite fit the button. In addition to this, clicking the text beneath the buttons on the home page does nothing, they should act like a link, taking the user to the cipher page they desire.

The code isn't as efficient as it could be, there's repeated code between JavaScript files, for example the copy function is written out four times, this is only needed once, all of the repeated code could have been placed in the same JavaScript file and referenced when required, this would have reduced the amount of code used. Although the code is pretty well optimized and looks neat. The code is split up into appropriate functions and the code is commented. The functions will be available to use for other applications in the future. Another method of writing ciphers includes the use of char codes, this eradicated the need for a function that returns the index of a char in the alphabet, although since three of the ciphers were built upon the Caesar cipher which used the function to return the index of a char, it was deemed unnecessary to program using char codes. Even though the char code method is more efficient.

On top of everything, the page doesn't re-size well, this website was designed for a screen of the dimensions 2560 x 1600, other resolutions weren't considered, this may arise some problems with other screen sizes, elements may be out of place or overlap.

5 Personal evaluation

From this coursework I have learned a lot about ciphers, I would say I now have an interest in ciphers, throughout the time I was writing the ciphers I was watching documentaries about ciphers, I would read articles about the history of ciphers and research which ciphers I would be able to implement into my website. I improved my programming skills with this coursework since the ciphers required a lot of logical thinking. I have improved a lot in my ability to write in HTML/CSS, I actively researched different ways of building a website and tried and tested many techniques.

I faced a few challenged with this coursework, I had never programmed in JavaScript before this coursework. To overcome this challenge I set aside a day in which I read through the w3schools - JavaScript tutorial, after this I was ready to start programming the ciphers. I initially used forms to store the input boxes, an error had occurred where the copy function wouldn't work on the forms, this took me a while to work out the problem, after researching the issue and trying to fix it myself I found that if I got rid of the form it would copy properly. I came to this conclusion after finding the tutorial on w3schools which shows how to copy text from an input box. A common error arose when testing the encryption and decryption of all the ciphers where the output would come up as "undefined". As it turned out, the issues were logic errors, to fix this problem, either the loops in the JavaScript code had to be re-tuned, or the variable scopes had to be updated, I found that making variables global solved a lot of errors, this was the main hurdle faced when writing the JavaScript.

I thought there was an issue with the Fractioned Morse cipher, certain letter pairs wouldn't encrypt properly, as it turns out, the code was working as it should, the fault is in the cipher its self and the way it works. After reviewing the JavaScript code, I found no error when compared to the cipher description. It was when I had discovered that the algorithm doesn't accept certain letter pairings that the issue was resolved.

When encrypting, there was an error where spaces were replaced with 8 spaces, the initial short term solution was to remove all white space when formatting the input, as shown in diagram 6. I had emailed the lecturer inquiring about this since it's not uncommon for ciphers to remove white space. It was preferred to have spaces present in the encrypted code simply to make the website more user friendly, since reading a sentence without any spaces can be confusing. The problem turned out to be that the *if statement* that checked for spaces was located inside the wrong for loop.

I feel like I performed well, although I could have done better, I wasn't challenged by the ciphers I wrote, the main issue I had was getting used to the programming language, but over time as I got more used to JavaScript, the smoother it became to write the ciphers. The finished website looks clean and easy on the eye, the information on each page is straight forward and the ciphers are usable. I believe the website looks really good, and I invested a lot of time to make it as user friendly as I could. All of the JavaScript is neat and pretty efficient, I got rid of a lot of unnecessary code. I challenged myself to be able to write four ciphers and have them all working properly, I achieved my goal and also created a user interface which is pleasing to add to this.

6 Refereneces

image 1

image 2

image 3

image 4

image 5

image 6

image 7

image 8

Diagram 1

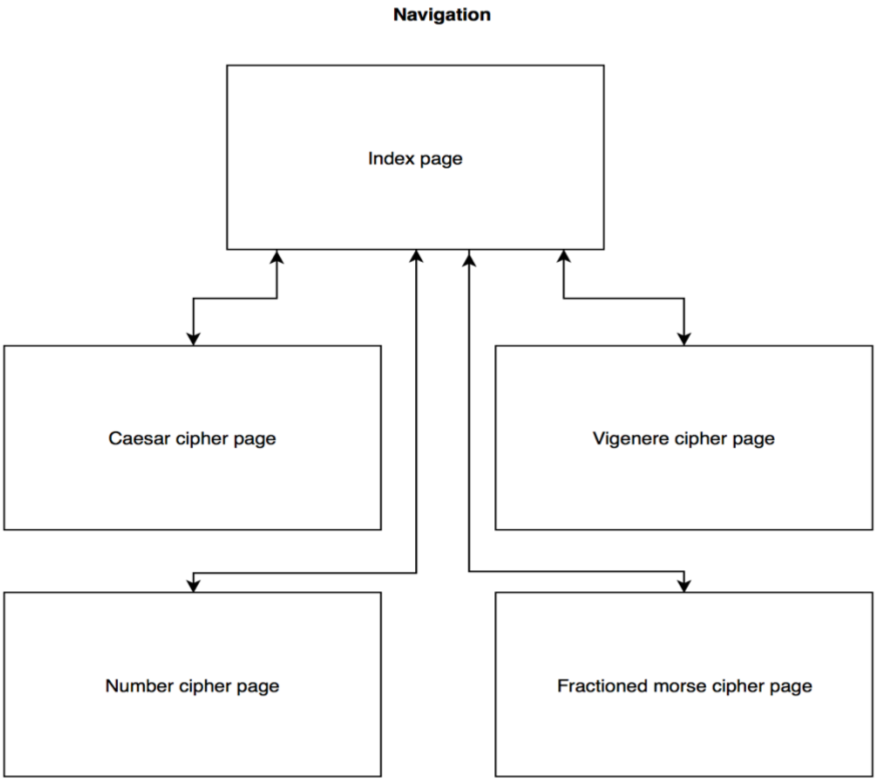


Diagram 2

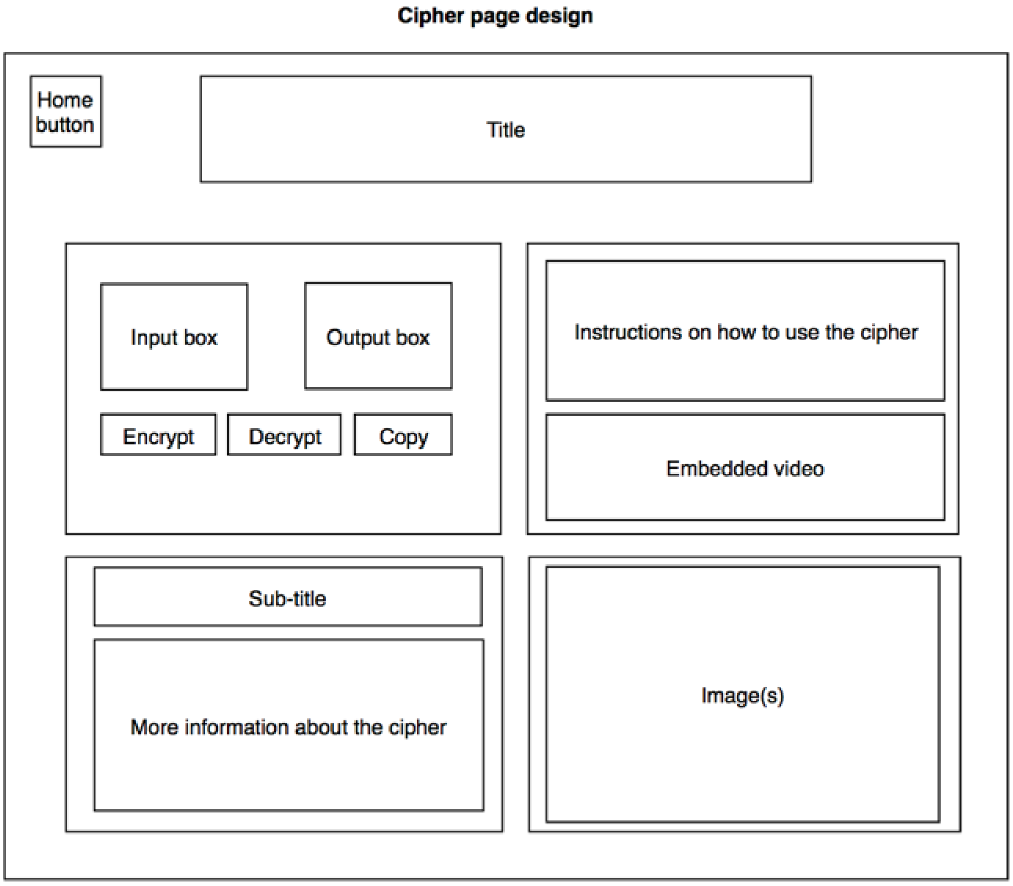


Diagram 3

Home page design

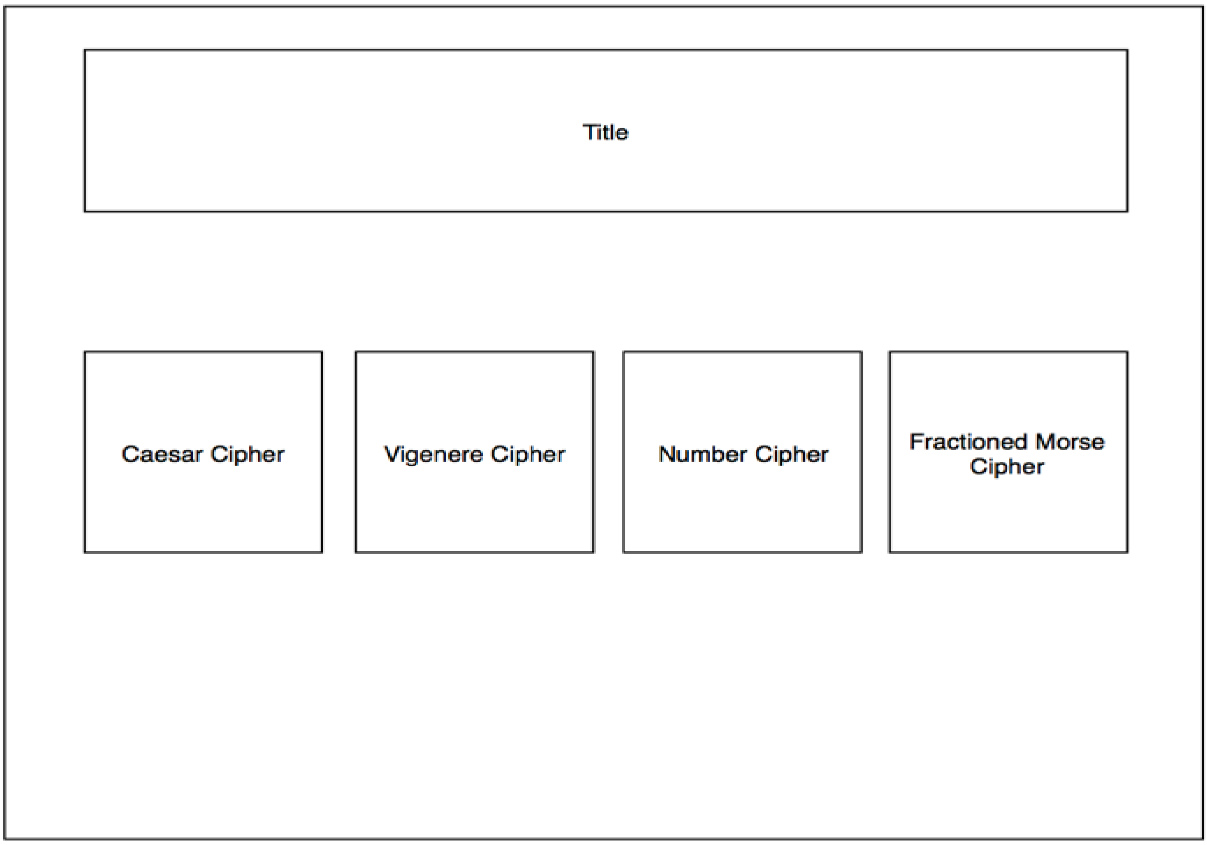
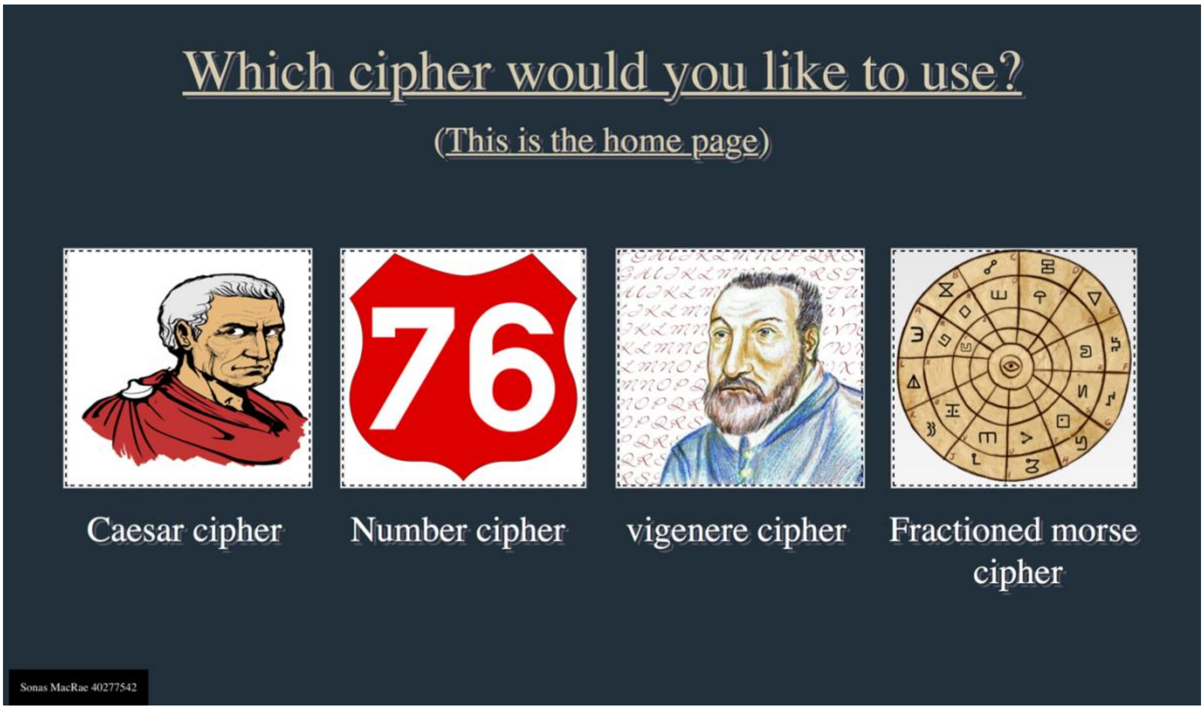


Diagram 4



Which cipher would you like to use?

(This is the home page)

Caesar cipher

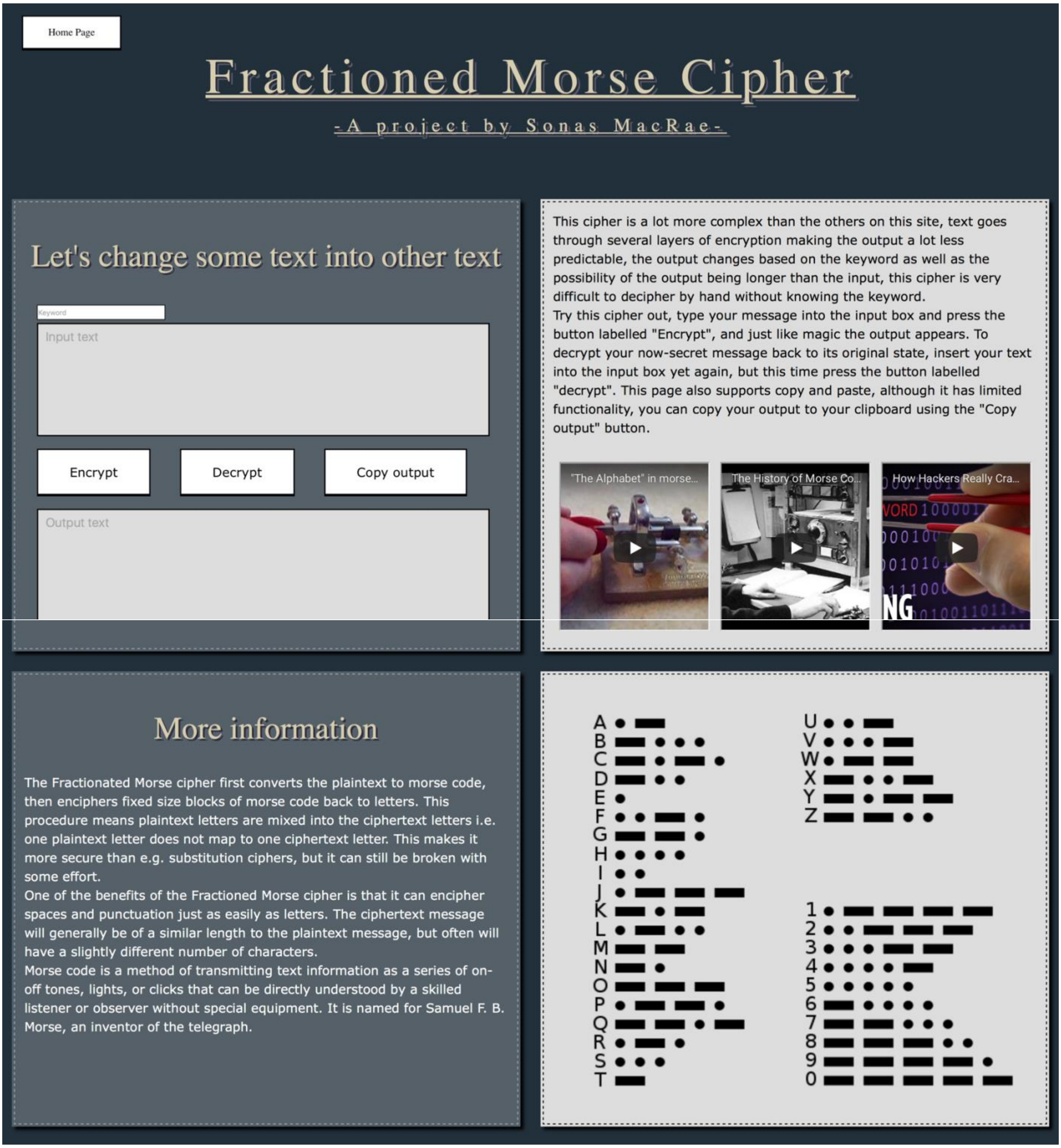
Number cipher

vigenere cipher

Fractioned morse cipher

Sonas MacRae 40277542

Diagram 5.1/5.2



Home Page

Fractioned Morse Cipher

-A project by Sonas MacRae-

Let's change some text into other text

Keyword

Input text

Encrypt | Decrypt | Copy output

Output text

This cipher is a lot more complex than the others on this site, text goes through several layers of encryption making the output a lot less predictable, the output changes based on the keyword as well as the possibility of the output being longer than the input, this cipher is very difficult to decipher by hand without knowing the keyword.
Try this cipher out, type your message into the input box and press the button labelled "Encrypt", and just like magic the output appears. To decrypt your now-secret message back to its original state, insert your text into the input box yet again, but this time press the button labelled "decrypt". This page also supports copy and paste, although it has limited functionality, you can copy your output to your clipboard using the "Copy output" button.

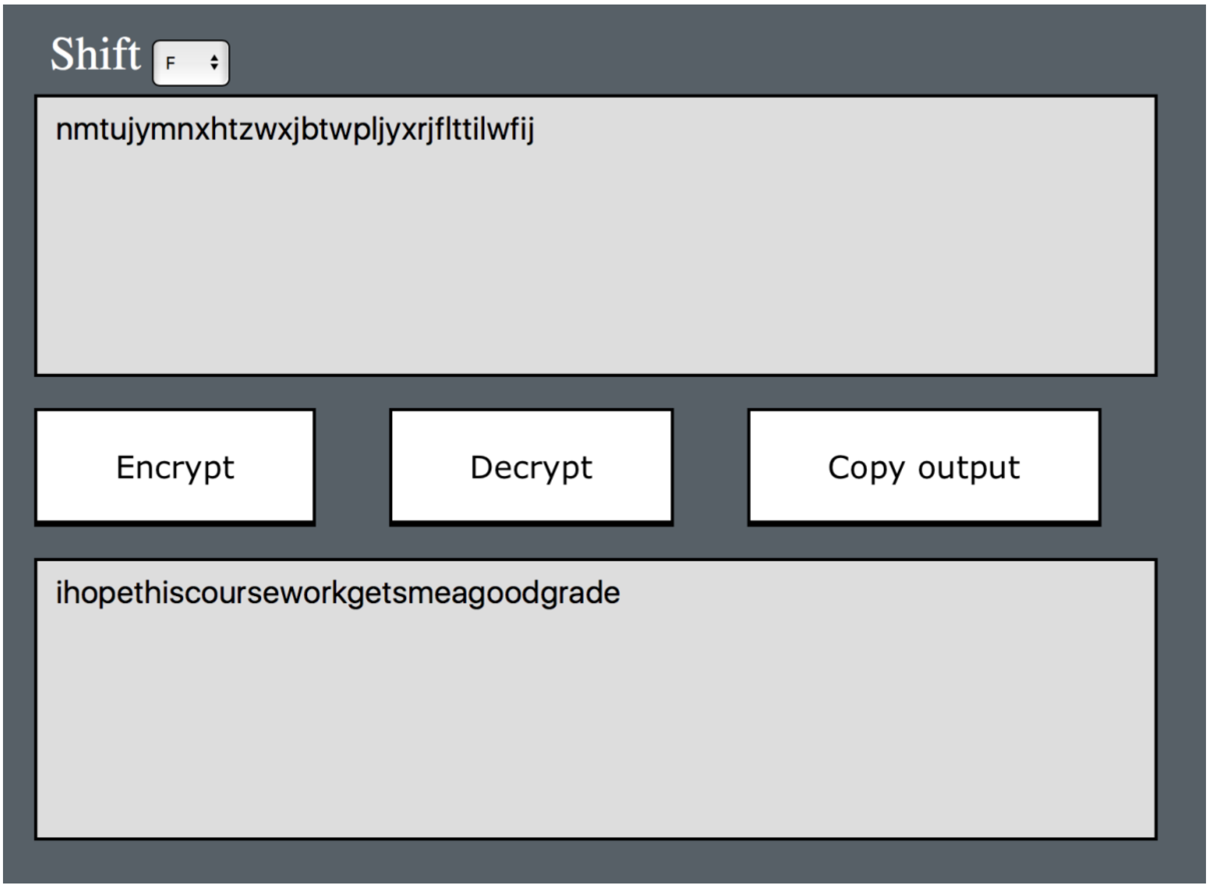
"The Alphabet" in morse...
The History of Morse Co...
How Hackers Really Cra...

More information

The Fractionated Morse cipher first converts the plaintext to morse code, then enciphers fixed size blocks of morse code back to letters. This procedure means plaintext letters are mixed into the ciphertext letters i.e. one plaintext letter does not map to one ciphertext letter. This makes it more secure than e.g. substitution ciphers, but it can still be broken with some effort.
One of the benefits of the Fractioned Morse cipher is that it can encipher spaces and punctuation just as easily as letters. The ciphertext message will generally be of a similar length to the plaintext message, but often will have a slightly different number of characters.
Morse code is a method of transmitting text information as a series of on-off tones, lights, or clicks that can be directly understood by a skilled listener or observer without special equipment. It is named for Samuel F. B. Morse, an inventor of the telegraph.

A	● ▬	U	● ● ▬
B	▬ ● ● ●	V	● ● ● ▬
C	▬ ● ▬ ●	W	● ▬ ▬
D	▬ ● ●	X	▬ ● ● ▬
E	●	Y	▬ ● ▬ ▬
F	● ● ▬ ●	Z	▬ ▬ ● ●
G	▬ ▬ ●		
H	● ● ● ●		
I	● ●		
J	● ▬ ▬ ▬	1	● ▬ ▬ ▬ ▬
K	▬ ● ▬	2	● ● ▬ ▬ ▬
L	● ▬ ● ●	3	● ● ● ▬ ▬
M	▬ ▬	4	● ● ● ● ▬
N	▬ ●	5	● ● ● ● ●
O	▬ ▬ ▬	6	▬ ● ● ● ●
P	● ▬ ▬ ●	7	▬ ▬ ● ● ●
Q	▬ ▬ ● ▬	8	▬ ▬ ▬ ● ●
R	● ▬ ●	9	▬ ▬ ▬ ▬ ●
S	● ● ●	0	▬ ▬ ▬ ▬ ▬
T	▬		

Diagram 6



Shift F

nmtujymnxhtzwxjbtwpljyxrjflttilwfij

ihopethiscourseworkgetsmeagoodgrade

Diagram 7

```
string input = sonas
array [] alphabet = [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]
for (int i = 0; i < alphabet. length; i++)
{
        for (int j = 0; j < input.length; j++)
        {
                if (input[j] == alphabet[i])
                {
                        int indexOfChar = i;
                }
        }
}
```