SET08101 Coursework 1 Report

Gabriele Balysevaite
40284966@live.napier.ac.uk
Edinburgh Napier University - Web Tech (SET00000)

Abstract

This was the first coursework of module SET08101 (Web Tech), it is worth 25 percent of our overall module grade. The aim of the coursework was to design and implement a website consisting of a set of pages about classical ciphers using HTML, CSS and JavaScript.

Keywords – cipher, Caesar, web, tech, SET08101, coursework, vigenere, ROT13, html, css, javascript

1 Introduction

I choose to implement four classical ciphers: Caesarian, ROT13, Vigenere and Rail Fence cipher. I started with Caesar and ROT13 ciphers, because they are simple substitution ciphers and having no previous experience with JavaScript before this module, it seemed like a good place to start. Afterwards I wanted to try to write a more challenging cipher, therefore I choose Vigenere cipher, a polyalphabetic substitution cipher. Lastly, I have decided to implement different kind of cipher, thus I wrote the Rail fence, a transposition cipher. I mostly relied on online resources on what kind of ciphers there are and how they work. Likewise, I have used online encoders-decoders to check whether my outputs are correct.

My web page has a simple, intuitive design. At the top there is a navigation bar with a 'Home' button and a button for every cipher I implemented. I selected contrasting colors for the navigation bar, header and the main body. There is aside content to describe what this web site is about. Also you can find a reference to a design page, with all typographical and presentational elements used in my site in the footer of every page.

2 Design implementation

2.1 Initial approach

First, I looked around the web for the best web page design at the moment, for example hubspot homepage designs and hubspot best website designs . Unfortunately, all the examples were for brands that were selling something and mostly included a lot of imagery of the product itself and the feeling, experience you get when you buy or use it. In my opinion web page about ciphers should be more oriented in providing information clearly and make it easily accessible, so you do not need to look at anything unrelated to what you were looking for. Having this in mind I have decided to make a minimalistic web page with no nonessential details in it.

2.2 Choosing layout

While looking at the HTML layout elements at w3schools I saw a simple layout example, that had header, footer navigation bar, aside element and main part containing section and article elements. (see Figure 1)

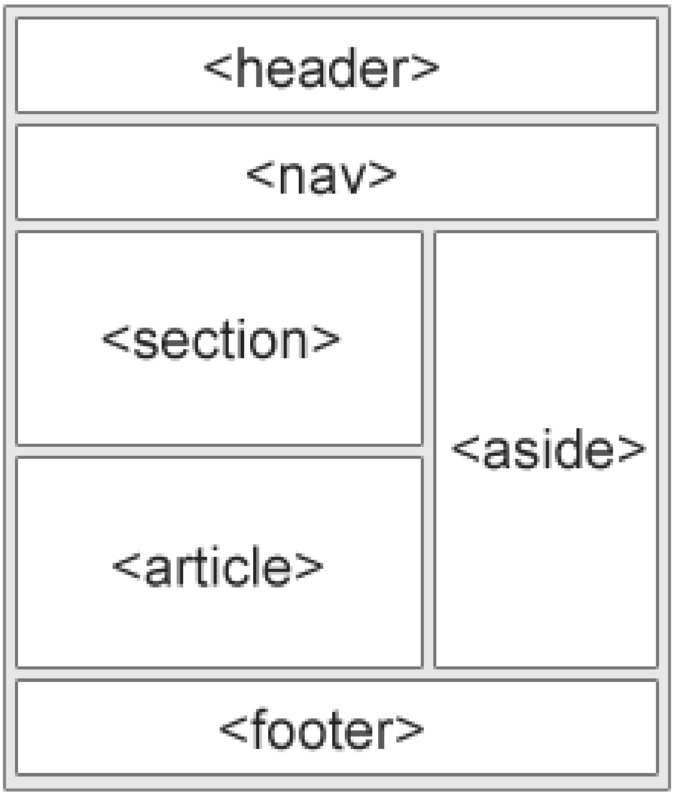


Figure 1: **Layout example** from w3shcools

The layout of the page w3school.com seemed like an appropriate layout for an educational web page and it was quite similar to the layout example offered at their HTML layout section, hence I decided to do something very similar to it. At first I tried to sketch my home page layout to see if I like how it would look for my page and decided on proportions of the elements.(see Figure 2)

I was pretty happy with this type of layout, because it felt practical and easy to navigate, I could easily change the number of ciphers I want to include and main body area would be scalable if I wish to include more text, examples or expand

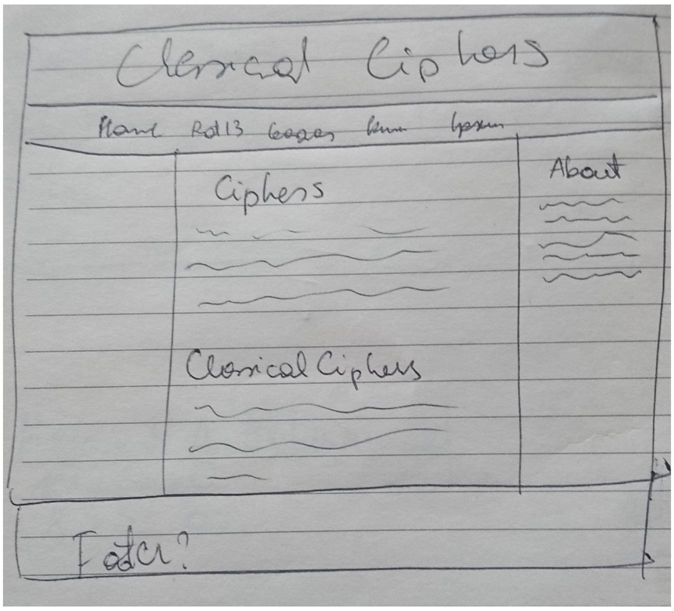
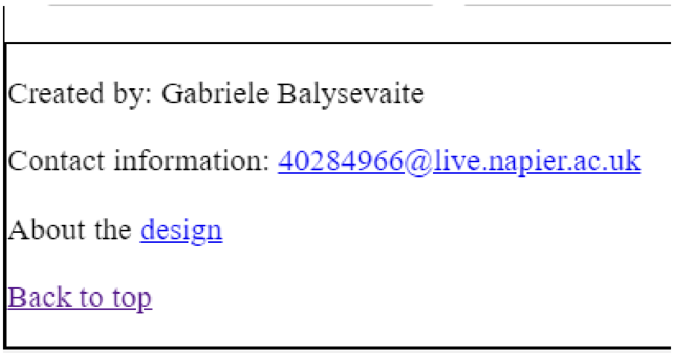


Figure 2: **First sketch** in the notebook

the text input and output areas.

Once I choose the layout, I started by simply putting header, navigation bar with "Ciper1", "Cipher2" etc. and a main body paragraph with some headers and placeholder text from Wikipedia. With CSS style document I added black borders to each element so I could easily see where one element ends and the other begins.(see Figure 3)

Then I played with margins, padding and text size, alignment to reach a more proportionate look. I set main body and aside element widths in percentage, so it would be suitable for different size screens. Also I set floats for the same elements to make them stay where I want without using tables or similar tools. I added back to top link in the footer, because I have seen it in many websites and I think it is a useful thing to have in a website layout when main part can go scroll down a lot. Also I added my university email and a link (not working at that time) to the design page. To keep web site minimal the top part (header1, navigation bar), aside and footer will stay the same through all the pages. (Figure 4)



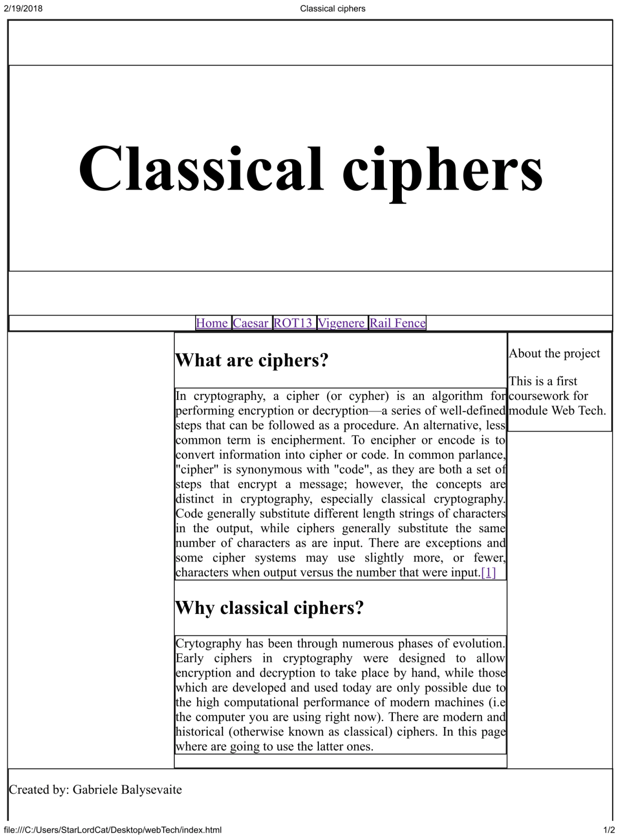
Created by: Gabriele Balysevaite

Contact information: 40284966@live.napier.ac.uk

About the design

Back to top

Figure 4: Footer of the page



2/19/2018 Classical ciphers

Classical ciphers

Home Caesar ROT13 Vigenere Rail Fence

What are ciphers?

In cryptography, a cipher (or cypher) is an algorithm for performing encryption or decryption—a series of well-defined steps that can be followed as a procedure. An alternative, less common term is encipherment. To encipher or encode is to convert information into cipher or code. In common parlance, "cipher" is synonymous with "code", as they are both a set of steps that encrypt a message; however, the concepts are distinct in cryptography, especially classical cryptography. Code generally substitute different length strings of characters in the output, while ciphers generally substitute the same number of characters as are input. There are exceptions and some cipher systems may use slightly more, or fewer, characters when output versus the number that were input.[1]

Why classical ciphers?

Crytography has been through numerous phases of evolution. Early ciphers in cryptography were designed to allow encryption and decryption to take place by hand, while those which are developed and used today are only possible due to the high computational performance of modern machines (i.e the computer you are using right now). There are modern and historical (otherwise known as classical) ciphers. In this page where are going to use the latter ones.

About the project

This is a first coursework for module Web Tech.

Created by: Gabriele Balysevaite

file:///C:/Users/StarLordCat/Desktop/webTech/index.html 1/2

Figure 3: Layout with borders

2.3 Choosing colours

Choosing a colour scheme was probably harder then choosing layout. Again, I browsed online and most of the pages about brands and colours were giving very similar information. I have decided to go with dark blue colour for as my dark colour because it is associated with trust and security. For my contrasting colour I wanted to go with some shade of purple, because it is represents wisdom or knowledge depending on the source. Having these two colours in mind I looked around in different colour schemes in coolors.co. However, there were no suitable schemes, which would look good for the website I imagined, therefore I tried to look for a shade of pink that would go together with shades of dark blue. At the end I ended up with colour scheme in Figure 5.

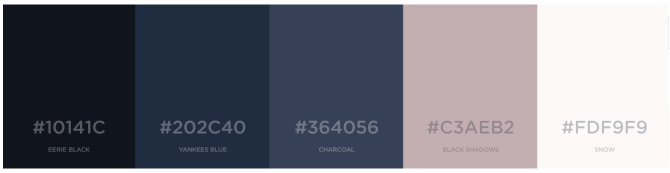


Figure 5: **Colours** final colour scheme

When I had the colours selected I needed to decide to make dark background and light text or light background and dark text. I concluded to go with the latter one, the reason being that: first - we were told during design lecture that it is a safer, easier to read way, secondly - I personally prefer websites that have light background colour and dark one for text.

The lightest colour "Snow" was set for the background of the main body and aside, with the colour of "Eerie black" for text. To separate header and footer from the body I used the colours opposite making the background dark and text light. These parts of the page do not contain a lot of text, so in my opinion it is acceptable to make the background dark with light text. When putting colours, the borders where deleted, because different colouring of elements will be enough to make further changes. To make the main body part stand out even more I made it colour of "Black shadows" (shade of pastel pink).

2.4 Improvements on design

With the right layout and colours the web page looked alright, but it was feeling a bit boring. Firstly, I changed the background of header 1. While looking I found that a moving image (gif) could work, and found dark coloured one. Then, I put a shadow on main body part and aside element, to create contrast between two similar coloured elements. For that I used CSS code from codepen.io. Lastly, I made navigation bar buttons change colour when mouse is hovered on them.

3 Implementation of ciphers

As mentioned before, the header, navigation bar, footer and aside are the same through all the pages. Therefore, I already had part of the pages done. First, to start implementing ciphers I had to use input and output areas. Initially I used ¡form¿ element for input and output, but later on I have changed them to ¡textarea¿ elements, because I could change the row number for it. For Caesar's and Rail fence ciphers a shift selection had to be added. For former 0 to 25 shift was available, but for latter one there was no limit. I decided to make Rail Fence shift up to 10, considering that it should be enough to show how the cipher is working. Likewise, Vigenere cipher required a input area for keyword. Lastly, I needed "Encrypt" and "Decrypt" buttons. I have decided to add "Clear all" button as well, because I believe it makes testing and use of the encrypting faster.

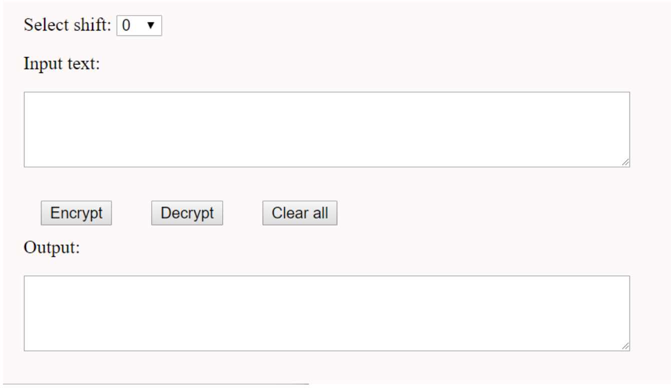


Figure 6: Input and output area

3.1 Caesar's cipher

To start I checked how the cipher is working using simple inputs on practicalcryptography.com. For both encryption and decryption, first I check if the character uppercase or lowercase letter, then using characters' ASCII code and the shift key value it is "pushed" to right for encryption and left for decryption. If the character is not a letter, it is just left as it is.

3.2 ROT13 cipher

ROT13 is a simple substitution cipher, where letter is replaced with 13th letter after it. For encryption, I checked if the letter is in first or second part of the alphabet, then, depending in which half the letter was, I either added (a to m) or subtracted (n to z) 13 to characters ASCII code value. Decryption is working in the same way, just subtraction and addition is switched places to reverse character to its' earlier value. Again nothing is done to non-alphabetical characters. To check if my implementation is correct is used rot13.com.

3.3 Vigenere cipher

Unsurprisingly this cipher was much harder to implement than others. I had to read quite a lot around to understand how it works and how I could make it work for me. First, for both encryption and decryption, I have checked if the keyword is present and if it contains any letters. This was done by converting keyword characters to ASCII code values and checking whether they are in the range of 65 to 122, then keyword is converted into array from 0 to 25, where each number represents the letter of alphabet. This is going to be the number of shift for the input letter in that space. The keyword is not case sensitive.

For decryption, keyword had to be put in "opposite" shift, for example if encrypting you pushed alphabet 11 places, to go back you will need to push back 26 - 11 = 15 places (the push and the change of letter).

The vigenere function takes each character from the given input, subtracts 65 for uppercase letters and 97 for lowercase letters from its ASCII code value, to make character code 0, then adds shift value from the key array and sets the value back to ASCII code. There is an integer, which goes up every time the replacement is done and makes keyword go from the beginning again and again, if the input is longer than the keyword. The code does not take non alphabetical characters to the key array and does not encrypt or decrypt them. I used counton.org to check if my result are correct.

3.4 Rail Fence cipher

Rail fence cipher does not work on key value of 0, therefore I did not put it as a choice. If the key of value 1 is chosen, the function returns the input itself. To encrypt plain text with rail fence cipher at first I created array with key number of empty arrays (arrays represent rails of the cipher). Code goes through each character of input and if the array was first one, the a char is pushed at the end of first array and a flag k is set to one representing the rail fence zigzag going down. If it is the last array, k value is set to -1 (because index of array is going to go "up" after), otherwise k flag is left the same and character is pushed to the array.

To decrypt encrypted text I created array with key number of empty arrays again. Then code goes through the input text in the list code puts a character from input into an array at the index "idx". Depending on the index of array (first, last or in between) the index "idx" is calculated in different ways. If it is the middle array, the index "idx" also depend on whether the letter is going into part of rail zigzag, that is going up or down. There is separate variable to keep track on that. After, all the elements of arrays are put into single output. I used crypto-online.net to check whether my answers were correct.

3.5 Finishing the website

After implementing all the ciphers, I put hyperlinks in every pages' HTML code, consequently making navigation between pages possible. At the end I did the design document trying to show all presentational and typographical elements used in creating this site.

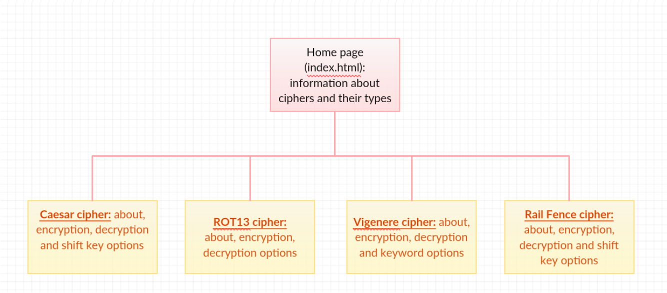

Home page (index.html): information about ciphers and their types
Caesar cipher: about, encryption, decryption and shift key options
ROT13 cipher: about, encryption, decryption options
Vigenere cipher: about, encryption, decryption and keyword options
Rail Fence cipher: about, encryption, decryption and shift key options

Figure 7: Navigation diagram

4 Evaluation

4.1 Requirements

There were only few distinctive requirements set in the descriptor of the coursework, therefore it was simple to fulfill them. We had to implement at least two ciphers and I choose to do two easy ones (Caesar's and ROT13), which we were familiarized with in practical sessions and two harder ones to challenge myself, but not to put too much work on my hands. I think I took appropriate work load for the coursework this weight. I did create different HTML documents for index (home) page, design document and one for each cipher, including text entry and display areas and JavaScript code document for encryption and decryption functions. I could not find any official or set requirements for the design of websites and at first it caused some confusion on how and where to start and what steps to follow. Nevertheless, I believed this showed me that website design is open for creativity and expression for brands as well as people.

4.2 Possible improvements

One of the things I think I could have done better is having a set of test inputs and output to help me faster test whether my code is working correctly. I used random input and that does not guarantee, that I checked all the possible cases for encryption and decryption. Second thing I would do differently next time is time committed to decide on the design of the page. This time I just choose the first option, that looked quite good. I feel I could have played more around with layouts and colour schemes, compare them to each other, maybe ask others, which one they prefer. That way I could find the best looking design not only for me, but also for others. Lastly, I realize I could have made the website more interactive. Perhaps adding some visuals on how the ciphers work, also making website act differently on various size screens, for example mobile and desktop screens.

5 References

Vayssouze-Faure F. (2017, July 31) Wavegrower-595x357.gif Retrieved 2018, February 13 from http://www.refinedguy.com/2017/07/31/wavegrower-hypnotic-animated-gifs-based-on-mathematical-concepts/

img_sem_elements.gif Retrieved 2018, February 16 from https://www.w3schools.com/html/html_layout.asp