Lab 6 C++

Deadline March 5th

Draft

Your name and student number must be at the top of every file.

You may use vector ,string and map classes in your code.

To Upload:

Upload all source files and data files you create.

Lab Book (pdf or doc)

Link to demo video. ( put in comments beside your name on top of every source file.)

Use onedrive ( or youtube) to store the video.

You must upload the code to blackboard before even if not all the questions are completed.

**You must upload a video link for the**  the lab and **upload the code** (and **Lab Book doc )**to get a mark.

Lab Book Diary

You must also show your lab book, when demoing the code.

The lab book shows your work as you do it and illustrates your approach to solving the problems below.

You must create a  **5 to 15 minute video** explaining the code and how you came about the solution or how you tried to come up with a solution.

You can use ObsStudio (https://obsproject.com/ ) to screen record. You should use your rough work and code to explain how you came about your solution.

Contents of Video

Running the code from visual Studio.

Showing the tests you used to check your code or a walkthrough of your code running.

Explain how you came about your code solution, supported with your code and your lab book. It may include approaches you tried but didn’t work.

**Caesar Cipher**

**Part 1 10%**

Implement a console version of the Caesar cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet.

<https://cryptii.com/pipes/caesar-cipher>

https://en.wikipedia.org/wiki/Caesar\_cipher

**Example**

*Plaintext*

this is a test for your code words are separated by one space and there is no punctuation the letters can be just lower case there are no numbers either just letters a to z

e.g.Replace with a letter 1 away from the input letter i.e a->b , b-> etc… z->a

*Ciphertext*

uijt jt b uftu gps zpvs dpef xpset bsf tfqbsbufe cz pof tqbdf boe uifsf jt op qvoduvbujpo uif mfuufst dbo cf kvtu mpxfs dbtf uifsf bsf op ovncfst fjuifs kvtu mfuufst b up a

Use a std::string to represent the plaintext. Only include the lowercase letters a to z, separated by a single space.

Allow the user to input how much to right shift each letter.

Output the ciphertext to the screen.

**Part 2: 10%**

**Code Breaker:**

Write code to read in the ciphertext file and ‘guess’ what the original text is:

Use the word.txt file in this folder on blackboard ( or find a better one) , add each word to a “map<string,bool>” variable that you create.

Given the ciphertext recreate the plaintext by shifting each letter to the left rather than the right.

e.g. a->z , b->a

The program must guess at the plaintext, it tries a value to left shift by and test if the resulting text yields many words that can be found in the map created form words.txt.

It outputs the decoded text to screen and the user selects ‘y’ or ‘n’ to indicate if the text has been decoded correctly.

If the user pressed ‘n’ then the program should come up with another ‘good’ guess.

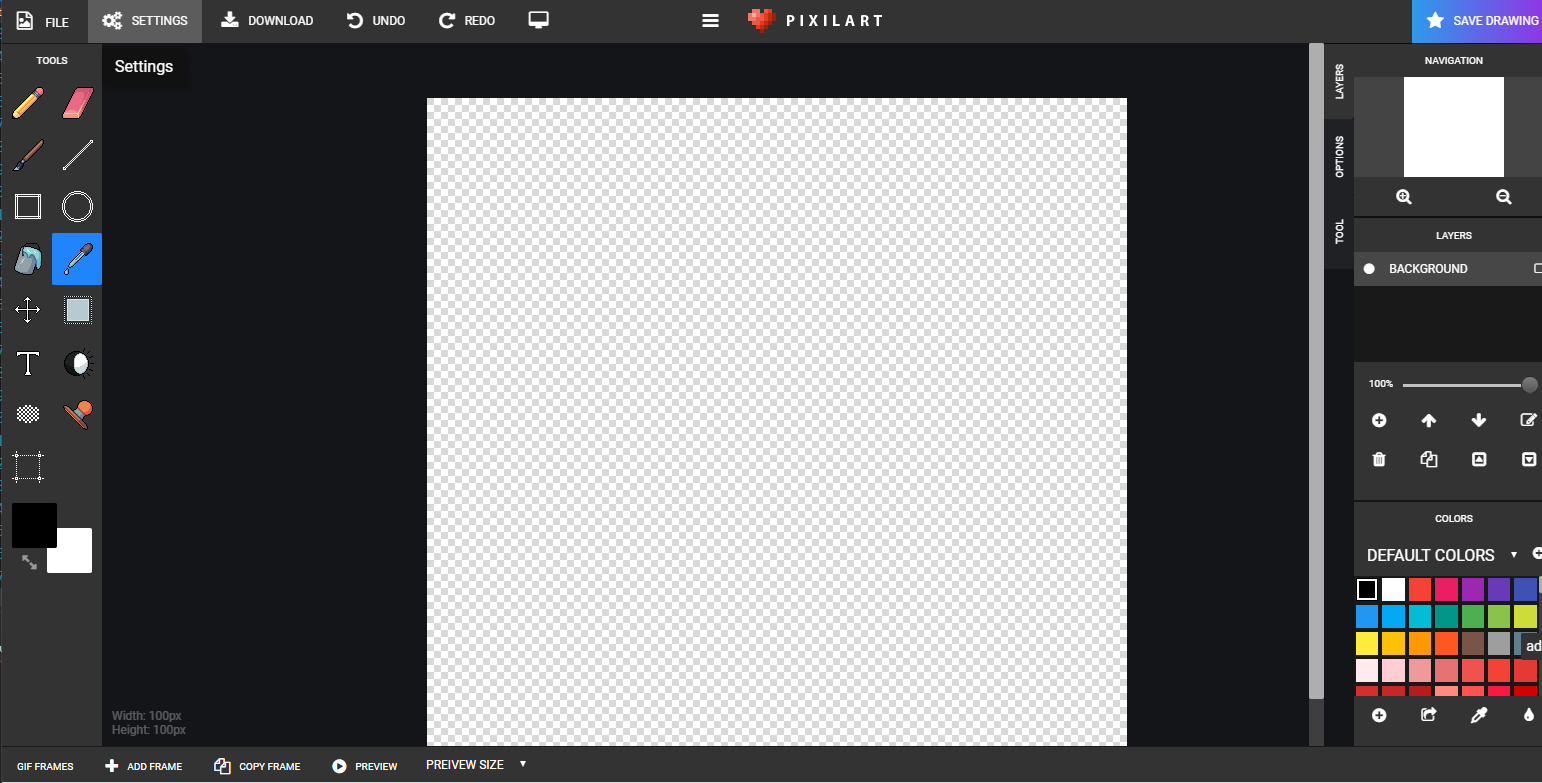
**Part 3: 10%**

**A better guess.**

Write code that reads in a text file (each word in the file is separated by 1 space, and contains only lowercase letters ) and it produces word frequency and letter frequency using stl::map.

Using letter frequency or word frequency from the above, write code that makes a guess on how much to shift the cyphertext in part 2 to produce the original plaintext .

**Part 4. Pixel Art Project 70%**



<https://www.pixilart.com/draw>

All must be your own work.

* Use the SFML Mouse Example.cpp as a starting point.
* Add buttons as in left of pixel art
* Add colour selection grid
* Add a drawing grid
* Add painting one block
* Add changing the colour with colour picker
* Adding eraser
* Add saving and loading
* Add drawing a rectangle
* Add selection tool with copy and paste.
* ……..