Requirements: python 3.8, PySide2

Obsługa

Ustawienia:

Ustawienia maszyny szyfrującej znajdują się w pliku config.json. Nieprawidłowe wprowadzenie danych skutkuje błędem podczas uruchamiania programu. Jeśli program nie wykryje pliku ustawień, sam stworzy plik domyślnych ustawień. Domyślne ustawienia odpowiadają konfiguracji modelowi Enigma I (1930).

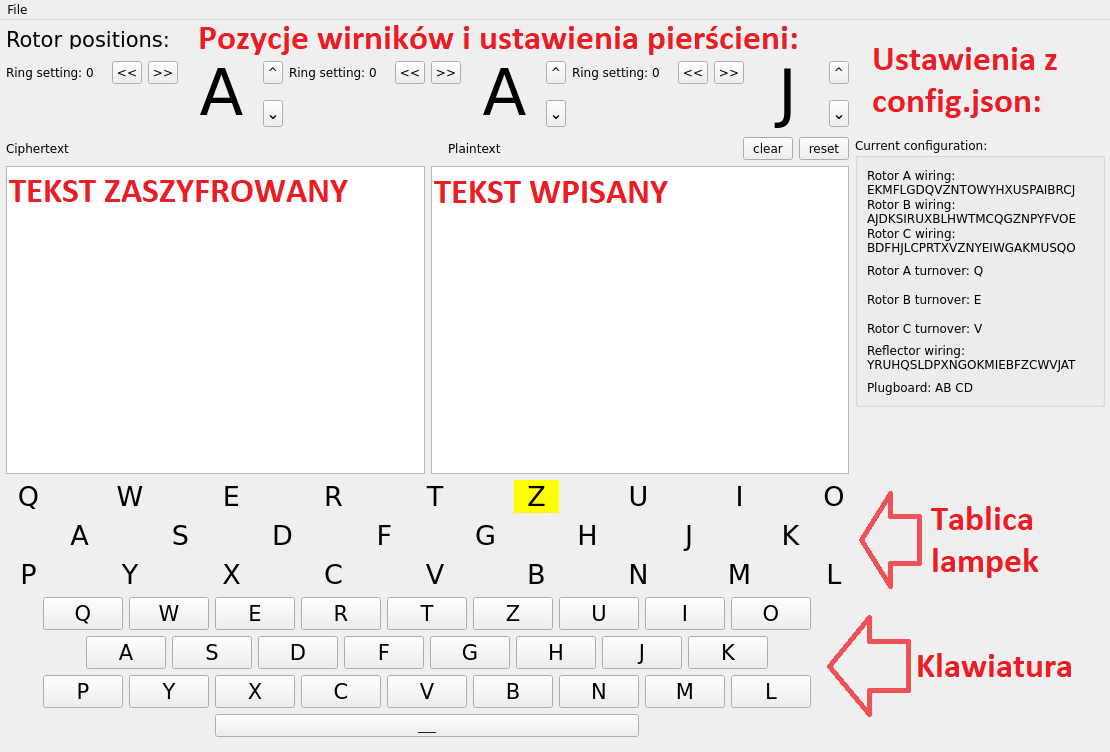
Poszczególne zmienne odpowiadają ustawieniom rzeczywistej maszyny szyfrującej:

rotor(Indeks wirnika)\_wiring – Połączenia między wejściami a wyjściami w wirniku. Litera wskazuje na wyjście połączenia, a pozycja litery, na wejście.   
Np. Dla „EKMFLGDQVZNTOWYHXUSPAIBRCJ” pierwsze połączenie, czyli wejścia 1, ma wyjście w 5, ponieważ litera E jest na 5 miejscu w angielskim alfabecie.  
Analogicznie działa ustawienie połączeń walca odwracającego(reflektora).  
Prawidłowo wprowadzone połączenie musi zawierać 26 wielkich liter angielskiego alfabetu. Litery te nie mogą się powtarzać.

Starting\_letter ustala pozycję w jakiej będzie znajdować się wirnik podczas uruchomienia programu.

Turnover specyfikuje, które litera jest pozycją przeskoku następnego wirnika. W rzeczywistej maszynie enigma, jeśli wirnik znajduje się w tej pozycji, to przy wprowadzeniu litery do maszyny, ten i następny wirnik muszą przejść o jedną pozycję.

Plugs specyfikuje połączenia w łącznicy kablowej. Prawidłowo wprowadzone połączenia muszą mieć postać [„litera1”, „litera2”]. Kolejność liter nie ma znaczenia. Nie można powtarzać tych samych połączeń.   
Przykładowo, prawidłowym połączeniem jest: [[„A”, „B”], [„C”, „D”]], w którym połączone są ze sobą litery A i B, oraz C i D.

Interfejs: 

Litery opisujące pozycję wirników odpowiadają literom ustawienia wirników w maszynie rzeczywistej. Strzałki pionowe pozwalają dowolnie je zmieniać. Ustawienia pierścienia mogą być skonfigurowane dla dowolnego wirnika strzałkami poziomymi.

Klawiatura i tablica lampek dla aplikacji działają analogicznie do tych w maszynie rzeczywistej. Możliwa jest obsługa za pomocą klawiatury komputera, oczywiście pamiętając, że zaszyfrowane mogą być jedynie litery angielskie, więc próba wpisania znaku „ś” przy użyciu alt+s, zakończy się wpisaniem „s”. Przycisk shift umożliwia wpisanie spacji do tekstu wpisanego jak i zaszyfrowanego.

Aplikacja pozwala zaszyfrować plik przy użyciu opcji File > Open. Wtedy plik tekstowy zostanie zaszyfrowany przy użyciu aktualnych ustawień, tych z config.json i tymczasowych, obowiązujących w interfejsie. Tekst zostanie sformatowany na możliwy do zaszyfrowania i automatycznie wpisany do maszyny. W ten sposób program może zaszyfrować pliki tekstowe do 5KB. Dla większych tekstów należy skorzystać z szyfrowania bezpośredniego File > Cipher directly.

Opcje save as… I save as… (only cipher) służą do zapisania szyfru i tekstu wpisanego. Opcja save as… zapisuje oba, a save as… (only cipher) tylko szyfr. Przy używaniu tych opcji należy pamiętać, że program nie zapamiętuje ustawień z jakimi zaczęliśmy szyfrować dany tekst.

Cipher directly pozwala bezpośrednio zaszyfrować plik tekstowy i go zapisać w innym pliku. W ten sposób można zaszyfrować do 10MB pliku tekstowego. Program zapisuje ustawienia tymczasowe interfejsu, z jakimi rozpoczęło się szyfrowanie, na końcu pliku. Warto pamiętać, że w enigmie szyfrowanie i odszyfrowanie przeprowadza się w dokładnie ten sam sposób.

Działanie

Moduły:

Implementacja projektu składa się z następujących modułów:

enigma\_cipher.py – zawiera metody szyfrowania tekstu i sprawdzania danych wejściowych. Dodatkowo przechowuje klasy reprezentujące błędy ustawień wejściowych.  
enigma\_classes.py – klasy przechowujące dane potrzebne do szyfrowania tekstu przy użyciu modułu enigma\_cipher.py. Klasy te zawierają dodatkowo metody zmiany swoich własnych wartości.  
enigma\_config\_io.py – posiada metodę pobierania ustawień z pliku json do programu.  
enigmaCipheringButton.py i enigmaLamp.py – posiadają klasy zaprojektowane do łatwiejszego programowania GUI.  
main.py – moduł główny, zawierający klasę okna aplikacji. Klasa te zawiera funkcje, których implementacja w innych modułach była utrudniona np. metody zapisy do pliku.