

VIGENERE

Wygenerowano przez Doxygen 1.8.17



# Rozdział 1

## Indeks plików

### 1.1 Lista plików

Tutaj znajduje się lista wszystkich udokumentowanych plików z ich krótkimi opisami:

C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">help.cpp</a> . . . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">help.h</a> . . . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">main.cpp</a> . . . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">mainProgramFunctions.cpp</a>	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">mainProgramFunctions.h</a> .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">parseArguments.cpp</a> . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">parseArguments.h</a> . . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">parseChars.cpp</a> . . . . .	??
C:/Users/jonny/source/WARUNEK_PPK/Projekt Vigenere/Vigenere/Vigenere/ <a href="#">parseChars.h</a> . . . . .	??



## Rozdział 2

# Dokumentacja plików

### 2.1 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/help.cpp

```
#include <iostream>
#include "help.h"
```

#### Funkcje

- void `programInstruction` ()

#### 2.1.1 Dokumentacja funkcji

##### 2.1.1.1 `programInstruction()`

```
void programInstruction ( )
```

Funkcja wyświetlająca instrukcję programu

### 2.2 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/help.h

#### Funkcje

- void `programInstruction` ()

## 2.2.1 Dokumentacja funkcji

### 2.2.1.1 programInstruction()

```
void programInstruction ( )
```

Funkcja wyświetlająca instrukcję programu

## 2.3 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/main.cpp

```
#include <iostream>
#include "mainProgramFunctions.h"
#include <vector>
```

### Funkcje

- int **main** (int amountOfArguments, char \*arguments[])

## 2.4 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/mainProgramFunctions.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include "parseArguments.h"
#include "parseChars.h"
#include "help.h"
```

### Funkcje

- bool **loadSwitches** (int howManyArguments, char \*arguments[], string &fileWithEncryptionKey, string &inputFile, string &outputFile, bool &encryption, bool &decryption, bool &breakingTheCode)
- vector< char > **loadDataToBuffer** (string inputFile)
- void **vigenere** (vector< char > &inputText, vector< char > &encryptedOrDecrypted, vector< char > &foundKey, string key, string outputFile, bool encryption, bool decryption, bool breakingTheKey)
- void **saveToFile** (vector< char > encryptedOrDecryptedText, vector< char > &foundKey, string &fileWithEncryptionKey, string outputFile, bool encryption, bool decryption, bool breakingTheKey)

## 2.4.1 Dokumentacja funkcji

### 2.4.1.1 loadDataToBuffer()

```
vector<char> loadDataToBuffer (
    string inputFile )
```

Funkcja wczytująca zawartość pliku do bufora

## Parametry

<i>inputFile</i>	nazwa pliku wejściowego
------------------	-------------------------

## 2.4.1.2 loadSwitches()

```
bool loadSwitches (
    int howManyArguments,
    char * arguments[],
    string & fileWithEncryptionKey,
    string & inputFile,
    string & outputFile,
    bool & encryption,
    bool & decryption,
    bool & breakingTheCode )
```

Funkcja wczytująca przełączniki podane przez wiersz poleceń

## Parametry

<i>howManyArguments</i>	liczba argumentów wczytana przez wiersz poleceń
<i>arguments</i>	nazwy argumentów wczytanych przez wiersz poleceń
<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>inputFile</i>	nazwa pliku zawierająca plik wejściowy
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheCode</i>	zmienna informująca o łamaniu klucza szyfrującego z pliku wejściowego

## 2.4.1.3 saveToFile()

```
void saveToFile (
    vector< char > encrypedOrDecrypedText,
    vector< char > & foundKey,
    string & fileWithEncryptionKey,
    string outputFile,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja zapisująca dane wyjściowe do pliku

## Parametry

<i>encrypedOrDecrypedText</i>	wektor zawierający zaszyfrowany lub zdeszyfrowany tekst wyjściowy
<i>foundKey</i>	wektor zawierający klucz szyfrujący



#### Parametry

<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

#### 2.4.1.4 vigenere()

```
void vigenere (
    vector< char > & inputText,
    vector< char > & encryptedOrDecrypted,
    vector< char > & foundKey,
    string key,
    string outputFile,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja wykonująca operacje(szyfrowanie/deszyfrowanie/łamanie szyfru)

#### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>encryptedOrDecrypted</i>	treść pliku wyjściowego
<i>foundKey</i>	treść znalezionej klucza szyfrującego
<i>key</i>	nazwa pliku zawierającego klucz szyfrujący
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

## 2.5 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/mainProgramFunctions.h

```
#include <iostream>
#include <vector>
```

### Funkcje

- bool [loadSwitches](#) (int howManyArguments, char \*arguments[], string &fileWithEncryptionKey, string &inputFile, string &outputFile, bool &encryption, bool &decryption, bool &breakingTheCode)

- `vector< char > loadDataToBuffer` (string inputFile)
- void `vigenere` (vector< char > &inputText, vector< char > &encryptedOrDecrypted, vector< char > &foundKey, string key, string outputFile, bool encryption, bool decryption, bool breakingTheKey)
- void `saveToFile` (vector< char > encryptedOrDecryptedText, vector< char > &foundKey, string &fileWithEncryptionKey, string outputFile, bool encryption, bool decryption, bool breakingTheKey)

## 2.5.1 Dokumentacja funkcji

### 2.5.1.1 loadDataToBuffer()

```
vector<char> loadDataToBuffer (
    string inputFile )
```

Funkcja wczytująca zawartość pliku do bufora

#### Parametry

<i>inputFile</i>	nazwa pliku wejściowego
------------------	-------------------------

### 2.5.1.2 loadSwitches()

```
bool loadSwitches (
    int howManyArguments,
    char * arguments[],
    string & fileWithEncryptionKey,
    string & inputFile,
    string & outputFile,
    bool & encryption,
    bool & decryption,
    bool & breakingTheCode )
```

Funkcja wczytująca przełączniki podane przez wiersz poleceń

#### Parametry

<i>howManyArguments</i>	liczba argumentów wczytana przez wiersz poleceń
<i>arguments</i>	nazwy argumentów wczytanych przez wiersz poleceń
<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>inputFile</i>	nazwa pliku zawierająca plik wejściowy
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheCode</i>	zmienna informująca o łamaniu klucza szyfrującego z pliku wejściowego

### 2.5.1.3 saveToFile()

```
void saveToFile (
    vector< char > encryptedOrDecryptedText,
    vector< char > & foundKey,
    string & fileWithEncryptionKey,
    string outputFile,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja zapisująca dane wyjściowe do pliku

#### Parametry

<i>encryptedOrDecryptedText</i>	wektor zawierający zaszyfrowany lub zdeszyfrowany tekst wyjściowy
<i>foundKey</i>	wektor zawierający klucz szyfrujący
<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

### 2.5.1.4 vigenere()

```
void vigenere (
    vector< char > & inputText,
    vector< char > & encryptedOrDecrypted,
    vector< char > & foundKey,
    string key,
    string outputFile,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja wykonująca operacje(szyfrowanie/deszyfrowanie/łamanie szyfru)

#### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>encryptedOrDecrypted</i>	treść pliku wyjściowego
<i>foundKey</i>	treść znalezionej klucza szyfrującego
<i>key</i>	nazwa pliku zawierającego klucz szyfrujący
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

## 2.6 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/parseArguments.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include "help.h"
#include "parseChars.h"
```

### Funkcje

- bool [checkSwitches](#) (int counter, char \*arguments[], string &fileWithEncryptionKey, string &inputFile, string &outputFile, bool &isInput, bool &isOutput, bool &isEncryptionKey)
- string [getlineFromFile](#) (string outputFile, string key, bool encryption, bool decryption, bool breakingTheKey)
- size\_t [checkSubstring](#) (const vector< char > &contentOfEncryptionKey, int iteration)
- vector< char > [extractEncryptionKey](#) (const vector< char > &contentOfEncryptionKey)

### 2.6.1 Dokumentacja funkcji

#### 2.6.1.1 checkSubstring()

```
size_t checkSubstring (
    const vector< char > & contentOfEncryptionKey,
    int iteration )
```

Funkcja sprawdzająca długość pod słowa

##### Parametry

<i>contentOfEncryptionKey</i>	treść klucza szyfrującego w którym sprawdzane są powtórzenia
<i>iteration</i>	aktualna iteracja pętli w funkcji nadrzędnej

#### 2.6.1.2 checkSwitches()

```
bool checkSwitches (
    int counter,
    char * arguments[],
    string & fileWithEncryptionKey,
    string & inputFile,
    string & outputFile,
    bool & isInput,
```

```
bool & isOutput,  
bool & isEncryptionKey )
```

Funkcja odpowiadająca za operacje na plikach dla podanych przełączników

#### Parametry

<i>counter</i>	aktualna iteracja pętli funkcji nadrzędnej
<i>arguments</i>	nazwy argumentów wczytanych przez wiersz poleceń
<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>inputFile</i>	nazwa pliku zawierająca plik wejściowy
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>isInput</i>	zmienna informująca o istnieniu pliku wejściowego
<i>isOutput</i>	zmienna informująca o istnieniu pliku wyjściowego
<i>isEncryptionKey</i>	zmienna informująca o istnieniu pliku z kluczem szyfrującym

#### 2.6.1.3 extractEncryptionKey()

```
vector<char> extractEncryptionKey (  
    const vector< char > & contentOfEncryptionKey )
```

Funkcja znajduje najdłuższe podśłowo w wektorze, na podstawie tego zwraca prawidłowy klucz szyfrujący

#### Parametry

<i>contentOfEncryptionKey</i>	treść klucza szyfrującego
-------------------------------	---------------------------

#### 2.6.1.4 getlineFromFile()

```
string getlineFromFile (  
    string outputFile,  
    string key,  
    bool encryption,  
    bool decryption,  
    bool breakingTheKey )
```

Funkcja wczytująca linię tekstu z pliku

#### Parametry

<i>outputFile</i>	nazwa pliku wyjściowego
<i>key</i>	nazwa pliku zawierającego klucz szyfrujący
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheCode</i>	zmienna informująca o łamaniu klucza szyfrującego z pliku wejściowego

## 2.7 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/parseArguments.h

```
#include <iostream>
```

### Funkcje

- bool [checkSwitches](#) (int counter, char \*arguments[], string &fileWithEncryptionKey, string &inputFile, string &outputFile, bool &isInput, bool &isOutput, bool &isEncryptionKey)
- string [getlineFromFile](#) (string outputFile, string key, bool encryption, bool decryption, bool breakingTheKey)
- size\_t [checkSubstring](#) (const vector< char > &contentOfEncryptionKey, int iteration)
- vector< char > [extractEncryptionKey](#) (const vector< char > &contentOfEncryptionKey)

### 2.7.1 Dokumentacja funkcji

#### 2.7.1.1 checkSubstring()

```
size_t checkSubstring (
    const vector< char > & contentOfEncryptionKey,
    int iteration )
```

Funkcja sprawdzająca długość pod słowa

##### Parametry

<i>contentOfEncryptionKey</i>	treść klucza szyfrującego w którym sprawdzane są powtórzenia
<i>iteration</i>	aktualna iteracja pętli w funkcji nadrzędnej

#### 2.7.1.2 checkSwitches()

```
bool checkSwitches (
    int counter,
    char * arguments[],
    string & fileWithEncryptionKey,
    string & inputFile,
    string & outputFile,
    bool & isInput,
    bool & isOutput,
    bool & isEncryptionKey )
```

Funkcja odpowiadająca za operacje na plikach dla podanych przełączników

#### Parametry

<i>counter</i>	aktualna iteracja pętli funkcji nadrzędnej
<i>arguments</i>	nazwy argumentów wczytanych przez wiersz poleceń
<i>fileWithEncryptionKey</i>	nazwa pliku zawierającego klucz szyfrujący
<i>inputFile</i>	nazwa pliku zawierająca plik wejściowy
<i>outputFile</i>	nazwa pliku zawierająca plik wyjściowy
<i>isInput</i>	zmienna informująca o istnieniu pliku wejściowego
<i>isOutput</i>	zmienna informująca o istnieniu pliku wyjściowego
<i>isEncryptionKey</i>	zmienna informująca o istnieniu pliku z kluczem szyfrującym

#### 2.7.1.3 extractEncryptionKey()

```
vector<char> extractEncryptionKey (  
    const vector< char > & contentOfEncryptionKey )
```

Funkcja znajduje najdłuższe podśłowo w wektorze, na podstawie tego zwraca prawidłowy klucz szyfrujący

#### Parametry

<i>contentOfEncryptionKey</i>	treść klucza szyfrującego
-------------------------------	---------------------------

#### 2.7.1.4 getlineFromFile()

```
string getlineFromFile (  
    string outputFile,  
    string key,  
    bool encryption,  
    bool decryption,  
    bool breakingTheKey )
```

Funkcja wczytująca linię tekstu z pliku

#### Parametry

<i>outputFile</i>	nazwa pliku wyjściowego
<i>key</i>	nazwa pliku zawierającego klucz szyfrujący
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheCode</i>	zmienna informująca o łamaniu klucza szyfrującego z pliku wejściowego

## 2.8 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/parseChars.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include "help.h"
#include "parseArguments.h"
```

### Funkcje

- int [checkKey](#) (string key)
- string [encryptionKeyShift](#) (string key)
- char [encryptChar](#) (unsigned char c, char shift)
- vector< char > [encrypt](#) (vector< char > inputText, const string &keyShifts)
- char [decryptChar](#) (unsigned char c, char shift)
- vector< char > [decrypt](#) (vector< char > inputText, const string &keyShifts)
- vector< char > [lookForTheKey](#) (vector< char > &inputText, string outputFile, string key, bool encryption, bool decryption, bool breakingTheKey)

### 2.8.1 Dokumentacja funkcji

#### 2.8.1.1 checkKey()

```
int checkKey (
    string key )
```

Funkcja sprawdzająca poprawność klucza szyfrującego

##### Parametry

<i>key</i>	klucz szyfrujący do sprawdzenia
------------	---------------------------------

#### 2.8.1.2 decrypt()

```
vector<char> decrypt (
    vector< char > inputText,
    const string & keyShifts )
```

Funkcja zwracająca tekst po deszyfrowaniu



#### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>keyShifts</i>	wartości o jakie należy przesunąć tekst wejściowy

#### 2.8.1.3 decryptChar()

```
char decryptChar (
    unsigned char c,
    char shift )
```

Funkcja deszyfrująca znaki

#### Parametry

<i>c</i>	litery z pliku wejściowego
<i>shift</i>	litery wskazujące o jaką wartość należy przesunąć litery w pliku wejściowym aby otrzymać litery w pliku wyjściowym

#### 2.8.1.4 encrypt()

```
vector<char> encrypt (
    vector< char > inputText,
    const string & keyShifts )
```

Funkcja zwracająca zaszyfrowany tekst

#### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>keyShifts</i>	wartości o jakie należy przesunąć tekst wejściowy

#### 2.8.1.5 encryptChar()

```
char encryptChar (
    unsigned char c,
    char shift )
```

Funkcja szyfrująca znaki

## Parametry

<i>c</i>	litery z pliku wejściowego
<i>shift</i>	litery wskazujące o jaką wartość należy przesunąć litery w pliku wejściowym aby otrzymać litery w pliku wyjściowym

**2.8.1.6 encryptionKeyShift()**

```
string encryptionKeyShift (
    string key )
```

Funkcja znajdująca wartości przesunięć liter w pliku wyjściowym na podstawie klucza szyfrującego

## Parametry

<i>key</i>	tekst względem którego został zakodowany plik wyjściowy
------------	---

**2.8.1.7 lookForTheKey()**

```
vector<char> lookForTheKey (
    vector< char > & inputText,
    string outputFile,
    string key,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja szukająca klucza szyfrującego

## Parametry

<i>inputText</i>	treść pliku wejściowego
<i>outputFile</i>	nazwa pliku wyjściowego
<i>key</i>	nazwa pliku z kluczem szyfrującym
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

## 2.9 Dokumentacja pliku C:/Users/jonny/source/WARUNEK\_PPK/Projekt Vigenere/Vigenere/Vigenere/parseChars.h

```
#include <iostream>
```

## Funkcje

- int `checkKey` (string key)
- string `encryptionKeyShift` (string key)
- char `encryptChar` (unsigned char c, char shift)
- vector< char > `encrypt` (vector< char > inputText, const string &keyShifts)
- char `decryptChar` (unsigned char c, char shift)
- vector< char > `decrypt` (vector< char > inputText, const string &keyShifts)
- vector< char > `lookForTheKey` (vector< char > &inputText, string outputFile, string key, bool encryption, bool decryption, bool breakingTheKey)

### 2.9.1 Dokumentacja funkcji

#### 2.9.1.1 `checkKey()`

```
int checkKey (  
    string key )
```

Funkcja sprawdzająca poprawność klucza szyfrującego

##### Parametry

<i>key</i>	klucz szyfrujący do sprawdzenia
------------	---------------------------------

#### 2.9.1.2 `decrypt()`

```
vector<char> decrypt (  
    vector< char > inputText,  
    const string & keyShifts )
```

Funkcja zwracająca tekst po deszyfrowaniu

##### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>keyShifts</i>	wartości o jakie należy przesunąć tekst wejściowy

### 2.9.1.3 decryptChar()

```
char decryptChar (
    unsigned char c,
    char shift )
```

Funkcja deszyfrująca znaki

#### Parametry

<i>c</i>	litery z pliku wejściowego
<i>shift</i>	litery wskazujące o jaką wartość należy przesunąć litery w pliku wejściowym aby otrzymać litery w pliku wyjściowym

### 2.9.1.4 encrypt()

```
vector<char> encrypt (
    vector< char > inputText,
    const string & keyShifts )
```

Funkcja zwracająca zaszyfrowany tekst

#### Parametry

<i>inputText</i>	treść tekstu wejściowego
<i>keyShifts</i>	wartości o jakie należy przesunąć tekst wejściowy

### 2.9.1.5 encryptChar()

```
char encryptChar (
    unsigned char c,
    char shift )
```

Funkcja szyfrująca znaki

#### Parametry

<i>c</i>	litery z pliku wejściowego
<i>shift</i>	litery wskazujące o jaką wartość należy przesunąć litery w pliku wejściowym aby otrzymać litery w pliku wyjściowym

### 2.9.1.6 encryptionKeyShift()

```
string encryptionKeyShift (
    string key )
```

Funkcja znajdująca wartości przesunięć liter w pliku wyjściowym na podstawie klucza szyfrującego

#### Parametry

<i>key</i>	tekst względem którego został zakodowany plik wyjściowy
------------	---

### 2.9.1.7 lookForTheKey()

```
vector<char> lookForTheKey (
    vector< char > & inputText,
    string outputFile,
    string key,
    bool encryption,
    bool decryption,
    bool breakingTheKey )
```

Funkcja szukająca klucza szyfrującego

#### Parametry

<i>inputText</i>	treść pliku wejściowego
<i>outputFile</i>	nazwa pliku wyjściowego
<i>key</i>	nazwa pliku z kluczem szyfrującym
<i>encryption</i>	zmienna informująca o szyfrowaniu pliku
<i>decryption</i>	zmienna informująca o deszyfrowaniu pliku
<i>breakingTheKey</i>	zmienna informująca o łamaniu klucza szyfrującego

