modAlphaCipher

1.0

Doxygen 1.9.1

Chapter 1

1.1

.

Chapter 2

2.1

.

Cipher
??
cipher_error
??

Chapter 3

3.1

.

Chapter 4

4.1 Cipher

.

```
#include <Cipher.h>
```

- Cipher ()=delete
  .
- Cipher (int w)
- wstring zakodirovat (Cipher w, wstring &s)
- wstring raskodirovat (Cipher w, wstring &s)

- wstring **getValidOpenText** (const std::wstring &s)
- wstring **getValidCipherText** (const std::wstring &s)
- int **getValidKey** (const int k, const std::wstring &s)

- int k
  , .

4.1.1

.

4.1.2 ()

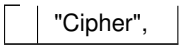
4.1.2.1 Cipher()

```
Cipher::Cipher (
            int w )
```

4.1.3

4.1.3.1 raskodirovat()

```
wstring Cipher::raskodirovat (
            Cipher w,
            wstring & s )
```

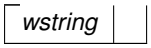


.

```
if (s.size()%w.k!=0) {
 h=s.size()/w.k+1;
 } else {
 h=s.size()/w.k;
 }
```

"wchar_t" , = "Cipher".

```
wchar_t a[h][w.k];
for (int i=0; i<w.k; i++) {
 for (int j=0; j<h; j++) {
     a[j][i]=s[k];
    k++;
     }
 }
```




```
for (int i=0; i<h; i++) {
  for (int j=0; j<w.k; j++) {
     decode+=a[i][j];
     }
 }
```

"wstring"

4.1.3.2 zakodirovat()

```
wstring Cipher::zakodirovat (
            Cipher w,
            wstring & s )
```

	"Cipher",
wstring	

.
```
if (s.size()%w.k!=0) {
   h=s.size()/w.k+1;
} else {
  h=s.size()/w.k;
} //   .
```

"wchar_t" , = "Cipher".
```
wchar_t a[h][w.k];
```

.
```
for (int i=0; i<h; i++) {
 for (int j=0; j<w.k; j++) {
     if (k<s.size()) {
         a[i][j]=s[k];
         k++;
     } else a[i][j]=' ';
 }
  }
for (int i=0; i<w.k; i++) {
 for (int j=0; j<h; j++) {
      code+=a[j][i];
  }
}
```

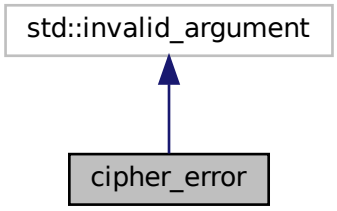
"wstring"

:

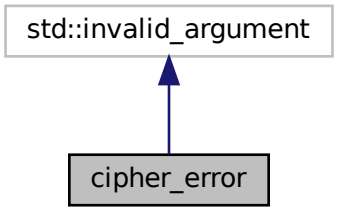
- Cipher.h

4.2 cipher_error

:cipher_error:



cipher_error:



- **cipher_error** (const std::string &what_arg)
- **cipher_error** (const char ∗what_arg)

4.2.1

:

- Cipher.h

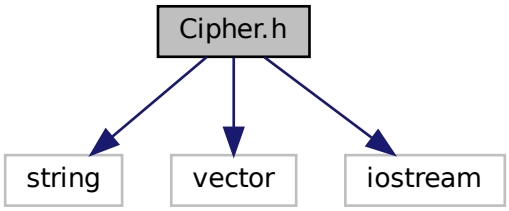
Chapter 5

5.1 Cipher.h

Cipher.

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

Cipher.h:



- class Cipher

  .
- class cipher_error

5.1.1

Cipher.

..

1.0

21.03.2022