

Отчет по лабораторной работе номер 1

Хамбалеев Булат Галимович

Содержание

1	Цель работы	5
2	Задание	6
3	Теория	7
4	Выполнение работы	8
5	Библиография	12
6	Выводы	13

List of Tables

List of Figures

4.1	рис.1. Аккаунт на на GitHub.	9
4.2	рис.2. Имя пользователя и почта.	10
4.3	рис.3. Создание ключа.	10
4.4	рис.4. Получение ключа.	10
4.5	рис.5. Ввод ключа.	11

1 Цель работы

Понять алгоритм шифра Цезаря и Атбаш.

2 Задание

Реализовать шифр Цезаря и шифр Атбаш.

3 Теория

Шифр Цезаря, также известный как шифр сдвига, код Цезаря — один из самых простых и наиболее широко известных методов шифрования. Шифр Цезаря — это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите. Шифр Атбаш является шифром сдвига на всю длину алфавита.

4 Выполнение работы

1. Реализуем класс Cesaг.(рис 1-3)


```

class Cesar():

    def __init__(self,k):
        num= int(k[:2])
        word = list(k[2:].strip().lower())
        c=0
        abc = list('абвгдеёжзийклмнопрстуфхцчщъыьэюя')
        abc_2 = ['_']*33
        for i in range(len(word)):
            if num+i>=33:
                abc_2[c]=word[i]
                c+=1
            else:
                abc_2[num+i]=word[i]

        for i in range(33):
            if abc_2[i]!='_' and abc_2[i+1]=='_':
                b =i+1
                break
        abc_3 = ''.join(abc)
        for letter in word:
            abc_3 = abc_3.replace(letter,'')
        c=0
        for i in range(33-len(word)):
            if b+i>=33:
                abc_2[c]=abc_3[i]
                c+=1
            else:
                abc_2[b+i]=abc_3[i]

        self.abc = dict(zip(abc,abc_2))
        self.abc.update({' ': ' '})
        self.bca = dict(zip(abc_2,abc))
        self.bca.update({' ': ' '})

```

Figure 4.1: рис.1. Аккаунт на на GitHub.

```

self.bca.update({letter: bca[letter]})

def code(self, text):
    return ''.join([self.abc[letter] for letter in text.lower()])
def decode(self, text):
    return ''.join([self.bca[letter] for letter in text.lower()])

```

Figure 4.2: рис.2. Имя пользователя и почта.

```

Ввод [86]: ces = Cesar('4 пароль')

Ввод [87]: ces.code('я гуляю')
Out[87]: 'ы якгыщ'

Ввод [88]: ces.decode('ы якгыщ')
Out[88]: 'я гуляю'

Ввод [89]: ces.code('Всем привет меня зовут Булат')
Out[89]: 'юиад жзьюай даеы лёюкй экгый'

Ввод [90]: ces.decode('юиад жзьюай даеы лёюкй экгый')
Out[90]: 'всем привет меня зовут булат'

```

Figure 4.3: рис.3. Создание ключа.

2. Реализуем класс Atbash.(рис 4-5)

```

6]: class Atbash():

    def __init__(self):

        abc = list('абвгдеёжзийклмнопрстуфхцшщъыьэюя ')
        abc_2 = abc[::-1]

        self.abc = dict(zip(abc, abc_2))
        self.bca = dict(zip(abc_2, abc))

    def code(self, text):
        return ''.join([self.abc[letter] for letter in text.lower()])
    def decode(self, text):
        return ''.join([self.bca[letter] for letter in text.lower()])

```

Figure 4.4: рис.4. Получение ключа.

```
Ввод [7]: atb = Atbash()

Ввод [8]: atb.code('меня зовут Булат')
Out[8]: 'уытбашсюмнаямф н'

Ввод [10]: atb.decode('уытбашсюмнаямф н')
Out[10]: 'меня зовут булат'

Ввод [5]: ''.join([' ', 'a'])
Out[5]: ' a'

Ввод [ ]:
```

Figure 4.5: рис.5. Ввод ключа.

5 Библиография

1. ТУИС РУДН

2. https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80_%D0%A6%D0%B5%D0%

6 Выводы

Во время выполнения лабораторной работы я освоил на практике шифр Цезаря и шифр Атбаш.