接口文档

SDES类

1. 构造函数

```
def __init__(self, key: List[int])
```

参数:

- `key` (List[int]): 一个包含10个二进制位的列表，表示S-DES的密钥。

2. encrypt方法

```
def encrypt(self, plaintext: List[int]) -> List[int]
```

参数:

- `plaintext` (List[int]): 一个包含8个二进制位的列表，表示要加密的明文。

返回值:

- `List[int]`: 一个包含8个二进制位的列表，表示加密后的密文。

3. decrypt方法

```
def decrypt(self, ciphertext: List[int]) -> List[int]
```

参数:

- `ciphertext` (List[int]): 一个包含8个二进制位的列表，表示要解密的密文。

返回值:

- `List[int]`: 一个包含8个二进制位的列表，表示解密后的明文。

4. encrypt_string方法

```
def encrypt_string(self, plaintext_str: str) -> str
```

参数:

- `plaintext_str` (str): 一个ASCII编码的字符串，表示要加密的明文。

返回值:

- `str`: 加密后的密文，以ASCII编码字符串的形式返回。

5. decrypt_string方法

```
def decrypt_string(self, ciphertext_str: str) -> str
```

参数:

- `ciphertext_str` (str): 一个ASCII编码的字符串，表示要解密的密文。

返回值:

- `str`: 解密后的明文，以ASCII编码字符串的形式返回。

5. 异常情况

- 如果输入的密钥、明文或密文的位数不正确，将引发`ValueError`异常。
- 如果在解密过程中遇到无效的输入，可能会引发解密失败的异常。

SDES破解

1. brute_force函数

```
def brute_force(pairs: List[Tuple[List[int], List[int]]]) ->
Tuple[Optional[List[int]], Optional[float]]
```

参数:

- `pairs` (List[Tuple[List[int], List[int]]]): 一个包含元组的列表，每个元组包含两个列表，分别表示明文和对应的密文。

返回值:

- `Tuple[Optional[List[int]], Optional[float]]`: 一个包含两个元素的元组。
  - 第一个元素是一个包含10个二进制位的列表，表示找到的密钥。如果没有找到密钥，则为`None`。
  - 第二个元素是一个浮点数，表示破解过程所花费的时间（以秒为单位）。

2. brute_force_all函数

```
def brute_force_all(pairs: List[List[List[int], List[int]]]) ->
Tuple[List[List[int]], float]
```

参数:

- `pairs` (List[List[List[int], List[int]]]):包含有明文，密文的列表。

返回值:

- `Tuple[List[List[int]], float]`: 一个包含两个元素的元组。
  - 第一个元素是一个包含列表的列表，每个列表表示找到的密钥。如果没有找到密钥，则为空列表。
  - 第二个元素是一个浮点数，表示破解过程所花费的时间（以秒为单位）。

为了开发的方便，我们也实现了Flask路由（接口），以下是该接口的介绍：

FLask路由

`/encrypt` 接口

功能:

该接口用于将Bit明文加密为密文。

HTTP请求方法:

- POST

输入参数:

- `key` (List[int]): 一个包含10个二进制位的列表，表示S-DES的密钥。
- `text` (List[int]): 一个包含8个二进制位的列表，表示要加密的明文。

输出格式:

- JSON对象，包含一个名为`result`的字段，其值是一个包含8个二进制位的列表，表示加密后的密文。

异常情况:

- 如果输入的密钥或明文的位数不正确，将返回HTTP 400 Bad Request响应。

`/decrypt` 接口

功能:

该接口用于将Bit密文解密为明文。

HTTP请求方法:

- POST

输入参数:

- `key` (List[int]): 一个包含10个二进制位的列表，表示S-DES的密钥。
- `text` (List[int]): 一个包含8个二进制位的列表，表示要解密的密文。

输出格式:

- JSON对象，包含一个名为`result`的字段，其值是一个包含8个二进制位的列表，表示解密后的明文。

异常情况:

- 如果输入的密钥或密文的位数不正确，将返回HTTP 400 Bad Request响应。

`/encrypt_ascii` 接口

功能:

该接口用于将ASCII编码的明文字符串加密为ASCII编码的密文字符串。

HTTP请求方法:

- POST

输入参数:

- `key` (List[int]): 一个包含10个二进制位的列表，表示S-DES的密钥。
- `text` (str): 一个ASCII编码的字符串，表示要加密的明文。

输出格式:

- JSON对象，包含一个名为`result`的字段，其值是一个ASCII编码的字符串，表示加密后的密文。

异常情况:

- 如果输入的密钥或明文字符串无效，将返回HTTP 400 Bad Request响应。

`/decrypt_ascii` 接口

功能:

该接口用于将ASCII编码的密文字符串解密为ASCII编码的明文字符串。

HTTP请求方法:

- POST

输入参数:

- `key` (List[int]): 一个包含10个二进制位的列表，表示S-DES的密钥。
- `text` (str): 一个ASCII编码的字符串，表示要解密的密文。

输出格式:

- JSON对象，包含一个名为`result`的字段，其值是一个ASCII编码的字符串，表示解密后的明文。

异常情况:

- 如果输入的密钥或密文字符串无效，将返回HTTP 400 Bad Request响应。

`/brute_force` 接口

功能:

该接口用于对给定的明文和密文对进行暴力破解，以找到S-DES的密钥。

HTTP请求方法:

- POST

输入参数:

- `pairs` (Dict): 一个字典，包含两个键值对，分别表示明文（`plaintext`）和密文（`ciphertext`）。

输出格式:

- JSON对象，包含以下字段：
  - `key` (List[int]): 一个包含10个二进制位的列表，表示找到的密钥。如果没有找到密钥，则为`null`。
  - `time_taken` (float): 一个浮点数，表示破解过程所花费的时间（以秒为单位）。

异常情况:

- 如果没有找到密钥，将返回HTTP 404 Not Found响应。

`/brute_force_all` 接口

功能:

该接口用于对给定的明文和密文对进行暴力破解，以找到所有可能的S-DES密钥。

HTTP请求方法:

- POST

输入参数:

- `pairs` (Dict): 一个字典，包含两个键值对，分别表示明文（`plaintext`）和密文（`ciphertext`）。

输出格式:

- JSON对象，包含以下字段:
  - `keys` (List[List[int]]): 一个包含列表的列表，每个列表表示找到的密钥。如果没有找到密钥，则为空列表。
  - `time_taken` (float): 一个浮点数，表示破解过程所花费的时间（以秒为单位）。

异常情况:

- 如果没有找到密钥，将返回HTTP 404 Not Found响应。