Homework 1: Substitution-Permutation Network

Name: Xuzheng Lu

GWid: G34363475

1. Environment and programming language

The algorirhm is written in Python (version = 3.7.4).

2. Format of the input file

The input file takes the format of `JSON`.

For better readability, the key and the x (or output) takes hexadecimal as input.

```
{
    "key": "1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c",
    "s-box": {
        "00": "63",
        ...
        "ff": "16"
    },
    "permutation function": [
        [0, 0],
        ...
        [31, 31]
    ],
    "m": 128,
    "x (or output)": "3a4b5c6d8e8e996678785299cbfdecab",
    "a": 0
}
```

Description:

"key": string, a 72-digit hexadecimal number (36 bytes)

"s-box": dict, the key is the hexadecimal of the 8 bits input number, and the value is the corresponding hexadecimal number of the output

"permutation function": 2D-list, the first element of each term in the list stands for the index of the input, and the second is the index of the output

"m": int, the length of the input in bits, should be a multiple of 32

"x (or output)": string, a hexadecimal number (m bits) of the input or output

"a": int, value 0 indicates encryption, and value 1 indicates decryption

3. Configuration

To set up the algorithm, you can modify the code at the end of the `spn.py` file as follows.

```
SPN(file_path='input_spn.txt', n_rounds=8, save_ouput=True).run()
```

Arguments:

file_path: str, default='input_spn.txt'. The path of the input file

n_rounds: int, default=8. The number of rounds of SPN

save_ouput: Boolean, default=True. If True, save the result to 'output_spn.txt'

4. Run the algorithm

You can run the algorithm by the following command.

```
python spn.py
```

Notice: If the default version of Python is 2.x, you should use the following command:

```
python3 spn.py
```

The algorithm will load `input_spn.txt`, and the output will be stored in `output_spn.txt`.

The result will also be displayed in the terminal (shell) as follows.

```
--------------------------------------------------------------------------------
Mode        : Encrytion
Key         : 1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c
Input text  : 3a4b5c6d8e8e996678785299cbfdecab
Onput cipher: 4b13ef56dd443a7d052acdb83e1cabcc
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Mode        : Decryption
Key         : 1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c1a2b3c4d5e6f7a8b9c
Input cipher: 4b13ef56dd443a7d052acdb83e1cabcc
Onput text  : 3a4b5c6d8e8e996678785299cbfdecab
--------------------------------------------------------------------------------
```