Bonus Lab

Implement a simplified version of the encryption algorithm in AES. In the simplified encryption algorithm, the `AddRoundKey` step in each round is skipped, i.e., only `SubBytes, ShiftRows, MixColumns` are considered. According to the AES, the `MixColumns` will not be applied in the last round (the 10th round).

You are allowed to hard-code plaintext in your program, and the plaintext must contain exactly 16 bytes of characters. You can consider the plaintext as the initial state, a 4x4 matrix. The state will then be updated by `SubBytes, ShiftRows, MixColumns` operations in each round. The S-Box and the matrix used for multiplication in `MixColumns` are provided in the next page. A sample output is provided as follows.

```
The plaintext is:
a b c d e f g h i j k l m n o p
-----------------------------------
After 1 round(s), the state is
BE B7 44 FB 59 77 FE 77 D1 CB ED E9 C4 48 8F C1
After 2 round(s), the state is
6E 9D 5E 34 D8 FA FA D0 79 0A 64 6F B9 7D 42 23
After 3 round(s), the state is
37 35 69 D8 26 B1 86 5F 5E C2 45 C3 98 CB CB 76
After 4 round(s), the state is
3A 93 E4 FA 9B 2A 8B E4 C6 44 A7 FD 63 47 B9 4F
After 5 round(s), the state is
F7 4B 7D 2B 31 A1 DC 7E 4A 5F 88 F5 31 FB 3D D4
After 6 round(s), the state is
0A CD 32 43 13 52 6C 09 11 B0 AA DA DE 3B 21 4E
After 7 round(s), the state is
4D DB 25 A9 A7 F9 5E 2F 55 64 00 AE 5B E1 39 55
After 8 round(s), the state is
F2 5C 45 08 93 07 F2 BC A3 79 DA 3F 27 1E 43 55
After 9 round(s), the state is
F6 AD 0B B3 1D 10 C8 48 FD 66 89 B5 F1 A1 39 0E
After 10 rounds (no MixColumn step), the state is
42 95 2B 6D CA E8 52 A4 A7 D5 54 33 AB A1 32 12
```

```
int[] sbox = { 0x63, 0x7C, 0x77, 0x7B, 0xF2, 0x6B, 0x6F,
            0xC5, 0x30, 0x01, 0x67, 0x2B, 0xFE, 0xD7, 0xAB, 0x76, 0xCA, 0x82,
            0xC9, 0x7D, 0xFA, 0x59, 0x47, 0xF0, 0xAD, 0xD4, 0xA2, 0xAF, 0x9C,
            0xA4, 0x72, 0xC0, 0xB7, 0xFD, 0x93, 0x26, 0x36, 0x3F, 0xF7, 0xCC,
            0x34, 0xA5, 0xE5, 0xF1, 0x71, 0xD8, 0x31, 0x15, 0x04, 0xC7, 0x23,
            0xC3, 0x18, 0x96, 0x05, 0x9A, 0x07, 0x12, 0x80, 0xE2, 0xEB, 0x27,
            0xB2, 0x75, 0x09, 0x83, 0x2C, 0x1A, 0x1B, 0x6E, 0x5A, 0xA0, 0x52,
            0x3B, 0xD6, 0xB3, 0x29, 0xE3, 0x2F, 0x84, 0x53, 0xD1, 0x00, 0xED,
            0x20, 0xFC, 0xB1, 0x5B, 0x6A, 0xCB, 0xBE, 0x39, 0x4A, 0x4C, 0x58,
            0xCF, 0xD0, 0xEF, 0xAA, 0xFB, 0x43, 0x4D, 0x33, 0x85, 0x45, 0xF9,
            0x02, 0x7F, 0x50, 0x3C, 0x9F, 0xA8, 0x51, 0xA3, 0x40, 0x8F, 0x92,
            0x9D, 0x38, 0xF5, 0xBC, 0xB6, 0xDA, 0x21, 0x10, 0xFF, 0xF3, 0xD2,
            0xCD, 0x0C, 0x13, 0xEC, 0x5F, 0x97, 0x44, 0x17, 0xC4, 0xA7, 0x7E,
            0x3D, 0x64, 0x5D, 0x19, 0x73, 0x60, 0x81, 0x4F, 0xDC, 0x22, 0x2A,
            0x90, 0x88, 0x46, 0xEE, 0xB8, 0x14, 0xDE, 0x5E, 0x0B, 0xDB, 0xE0,
            0x32, 0x3A, 0x0A, 0x49, 0x06, 0x24, 0x5C, 0xC2, 0xD3, 0xAC, 0x62,
            0x91, 0x95, 0xE4, 0x79, 0xE7, 0xC8, 0x37, 0x6D, 0x8D, 0xD5, 0x4E,
            0xA9, 0x6C, 0x56, 0xF4, 0xEA, 0x65, 0x7A, 0xAE, 0x08, 0xBA, 0x78,
            0x25, 0x2E, 0x1C, 0xA6, 0xB4, 0xC6, 0xE8, 0xDD, 0x74, 0x1F, 0x4B,
            0xBD, 0x8B, 0x8A, 0x70, 0x3E, 0xB5, 0x66, 0x48, 0x03, 0xF6, 0x0E,
            0x61, 0x35, 0x57, 0xB9, 0x86, 0xC1, 0x1D, 0x9E, 0xE1, 0xF8, 0x98,
            0x11, 0x69, 0xD9, 0x8E, 0x94, 0x9B, 0x1E, 0x87, 0xE9, 0xCE, 0x55,
            0x28, 0xDF, 0x8C, 0xA1, 0x89, 0x0D, 0xBF, 0xE6, 0x42, 0x68, 0x41,
            0x99, 0x2D, 0x0F, 0xB0, 0x54, 0xBB, 0x16 };


    int[] tran_matrix = {2, 3, 1, 1, 1, 2, 3, 1, 1, 1, 2, 3, 3, 1, 1, 2 };
```