CS Assignment 1 TA39
Code:

```
def p10_permutation(key):
    p10_table = [3, 5, 2, 7, 4, 10, 1, 9, 8, 6]
    p10_result = [key[i - 1] for i in p10_table]
    return ''.join(p10_result)

def p8_permutation(key):
    p8_table = [6, 3, 7, 4, 8, 5, 10, 9]
    p8_result = [key[i - 1] for i in p8_table]
    return ''.join(p8_result)

def left_shift(key, num_shifts):
    return key[num_shifts:] + key[:num_shifts]

def generate_subkeys(key):
    key_after_p10 = p10_permutation(key)
    left_key_half = key_after_p10[:5]
    right_key_half = key_after_p10[5:]
    left_shifted_key1 = left_shift(left_key_half, 1)
    left_shifted_key2 = left_shift(right_key_half, 1)
    subkey1 = p8_permutation(left_shifted_key1 + left_shifted_key2)
    left_shifted_key3 = left_shift(left_shifted_key1, 2)
    left_shifted_key4 = left_shift(left_shifted_key2, 2)
    subkey2 = p8_permutation(left_shifted_key3 + left_shifted_key4)
    return subkey1, subkey2

if __name__ == "__main__":
    key = input("Enter a 10-bit key (binary): ")
    if len(key) != 10 or not all(bit in '01' for bit in key):
        print("Invalid input. Please enter a 10-bit key consisting of 0s and 1s.")
    else:
        subkey1, subkey2 = generate_subkeys(key)
        print("Subkey 1:", subkey1)
        print("Subkey 2:", subkey2)
```

Output:

```
Enter a 10-bit key (binary): 1010000010
Subkey 1: 10100100
Subkey 2: 01000011
```