Mini Project -2 (FLAMES)
(Language PYTHON)

- It is a popular game name after the acronym , Name , Friend , Love , Affection , Enemies and siblings this game doesn't accrual predict whether or not an individual is for you.
    1. Take two names
    2. Remove the common characters
    3. Get the count of the characters that are left
    4. Take FLAMES letters F,L,A,M,E,S
    5. Start removing letter using the count we got
    6. The letter which last the process is the result

- **Input Format**: The input should be of two lines , player 1 name and player 2 name .
- **Output Format :** Relationship states is:__

Steps Involved:

1. **Input Processing:**
    - The user inputs two names.
    - Both names are converted to lowercase and stripped of spaces.
    - Each name is then converted into a list of characters.

2. **Removing Common Characters:**
    - The remove_match_char function is used to identify and remove common characters from the two lists of characters derived from the names.
    - If common characters are found, they are removed from both lists.
    - The remaining characters from both lists are concatenated with a special border marker * separating them.
    - The function returns this concatenated list along with a flag indicating whether any characters were removed.
    -

3. **Iterative Processing:**
    - The code repeatedly calls remove_match_char until no more common characters are left.
    - It updates the character lists each time, continuing until the lists have no common characters.

4. **Count Calculation:**
    - Once there are no common characters, the code calculates the total number of remaining characters.

5. **FLAMES Calculation:**
    - A predefined list of relationship statuses (["Friends", "Love", "Affection", "Marriage", "Enemy", "Siblings"]) is used to determine the final result.
    - Based on the count of remaining characters, the list is reduced in a circular manner.
    - The reduction process involves removing an element based on the count modulo the length of the list until only one element remains.

6. **Output:**
    - The final relationship status (the remaining element in the FLAMES list) is printed as the result.

Key Functions and Components:

- remove_match_char(list1, list2): Function to remove common characters between two lists and return the modified lists along with a flag.
- Iterative Loop: Continuously processes the lists to remove common characters until no common characters remain.
- FLAMES Calculation Loop: Uses a modular arithmetic approach to cyclically reduce the list of relationship statuses based on the total count of remaining characters.

Pseudo code:

```
Function remove_match_char(list1, list2):
    For each character in list1:
        For each character in list2:
            If the characters are the same:
                Store the character in variable c
                Remove the character from list1
                Remove the character from list2
                Concatenate list1, ["*"], and list2 into list3
                Return [list3, True]
    Concatenate list1, ["*"], and list2 into list3
    Return [list3, False]


Start:
    Input Player 1 name as p1
    Convert p1 to lowercase
    Remove spaces from p1
    Convert p1 to a list of characters p1_list

    Input Player 2 name as p2
    Convert p2 to lowercase
    Remove spaces from p2
    Convert p2 to a list of characters p2_list
```

```
Set proceed to True

While proceed is True:
    Call remove_match_char with p1_list and p2_list
    Store the result in ret_list
    Extract list3 from ret_list
    Extract flag from ret_list
    Set proceed to the flag value
    Find the index of "*" in list3
    Update p1_list to all characters before "*"
    Update p2_list to all characters after "*"

Calculate count as the sum of lengths of p1_list and p2_list

Initialize result with ["Friends", "Love", "Affection", "Marriage", "Enemy", "Siblings"]

While the length of result is greater than 1:
    Calculate split_index as (count % length of result) - 1
    If split_index is greater than or equal to 0:
        Split result into two lists:
            right = elements after split_index
            left = elements before split_index
        Concatenate right and left into result
    Else:
        Remove the last element from result

Print "Relationship status: " followed by the first element in result
```

Code for Number Guessing Game:

```
1. def remove_match_char(list1, list2): for i in range(len(list1)):
2. for j in range(len(list2)): if list1[i] == list2[j]:
3. c = list1[i] list1.remove(c) list2.remove(c)
4. list3 = list1 + ["*"] + list2 return [list3, True]
5. list3 = list1 + ["*"] + list2 return [list3, False]
6.
7. if    name         == " main   ":
8. p1 = input("Player 1 name : ") p1 = p1.lower()
9. p1 = p1.replace(" ", "") p1_list = list(p1)
10.
11. p2 = input("Player 2 name : ") p2 = p2.lower()
12. p2 = p2.replace(" ", "") p2_list = list(p2)
13. proceed = True while proceed:
14. ret_list = remove_match_char(p1_list, p2_list)
15. con_list = ret_list[0] proceed = ret_list[1]
16. star_index = con_list.index("*") p1_list = con_list[: star_index] p2_list = con_list[star_index +
1:]
17.
18. count = len(p1_list) + len(p2_list)
19. result = ["Friends", "Love", "Affection", "Marriage", "Enemy", "Siblings"]
20.
21. while len(result) > 1:
22. split_index = (count % len(result) - 1) if split_index >= 0:
23. right = result[split_index + 1:] left = result[: split_index] result = right + left
24. else:
25. result = result[: len(result) - 1]
26.  print("Relationship status :", result[0])
```

Output:

```
PS D:\py> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe d:/py/flames.py
Player 1 name : ramu
Player 2 name : rama
Relationship status : Enemy
PS D:\py> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe d:/py/flames.py
Player 1 name : rahil
Player 2 name : sonia
Relationship status : Marriage
PS D:\py> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe d:/py/flames.py
Player 1 name : ajay
Player 2 name : priya
Relationship status : Friends
PS D:\py> & C:/Users/ASUS/AppData/Local/Programs/Python/Python312/python.exe d:/py/flames.py
Player 1 name : jaret
Player 2 name : chole
Relationship status : Affection
```