EXPERIMENT 4.1.1

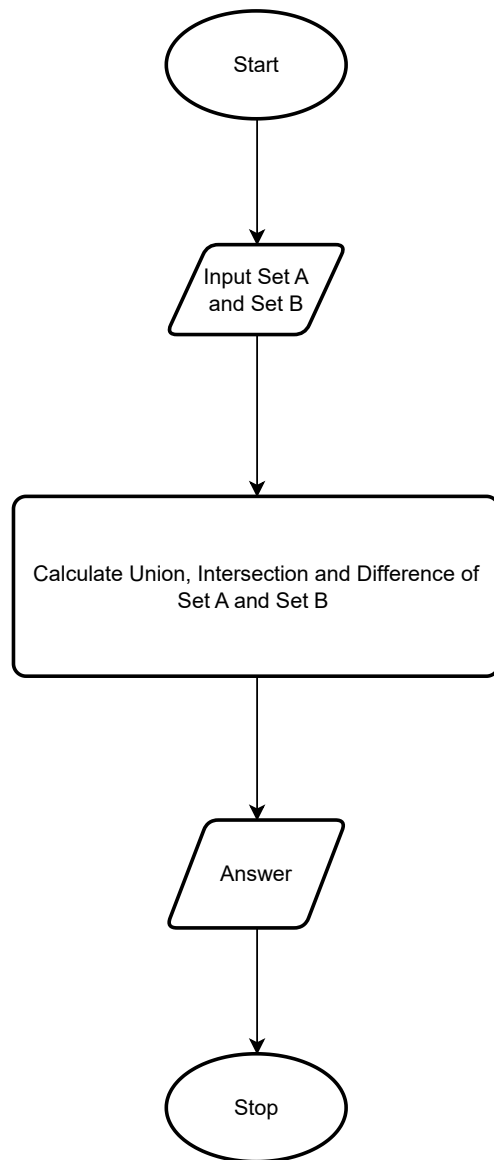**Algorithm:**

Step 1: Start.
Step 2: Input Set A and Set B.
Step 3: Calculate Intersection, Union and Difference of Set A and Set B.
Step 4: Display Intersection, Union and Difference.
Step 5: Stop.

Code:

```
Set_A = set(map(int, input("Set A: ").split()))
Set_B = set(map(int, input("Set B: ").split()))
union_set = Set_A | Set_B
intersection_set = Set_A & Set_B
difference_set = Set_A - Set_B
print(f"Union: {union_set}")
print(f"Intersection: {intersection_set}")
print(f"Difference: {difference_set}")
```

FlowChart:

```
        ┌─────────────┐
        │    Start    │
        └─────────────┘
               │
               ▼
         ┌──────────────┐
         │ Input Set A  │
         │  and Set B   │
         └──────────────┘
               │
               ▼
    ┌──────────────────────────────────────┐
    │ Calculate Union, Intersection and      │
    │ Difference of Set A and Set B          │
    └──────────────────────────────────────┘
               │
               ▼
         ┌──────────────┐
         │   Answer     │
         └──────────────┘
               │
               ▼
        ┌─────────────┐
        │    Stop     │
        └─────────────┘
```

4.1.1. Set Operations

Write a Python program to perform union, intersection and difference operations on *Set A* and *Set B*.

Input Format:
- First Line prompts "Set A: " followed by space-separated list of integers for *Set A*.
- The second input prompts "Set B: " followed by space-separated list of integers for *Set B*.

Output Format:
- The first line prints "Union: " followed by the union of *Set A* and *Set B*.
- The second line prints "Intersection: " followed by the intersection of *Set A* and *Set B*.
- The third line prints "Difference: " followed by the difference of *Set A* and *Set B*.

Note:
- If there is no intersection between the two sets, the program prints an empty set, which appears as "set()" in the output.
- Please refer to the visible test cases for better understanding.

Sample Test Cases   +

setoperat...

```python
Set_A = set(map(int, input("Set A: ").split()))
Set_B = set(map(int, input("Set B: ").split()))

union_set = Set_A | Set_B
intersection_set = Set_A & Set_B
difference_set = Set_A - Set_B
print(f"Union: {union_set}")
print(f"Intersection: {intersection_set}")
print(f"Difference: {difference_set}")
```

Average time	Maximum time
0.027 s	**0.080 s**
27.25 ms	80.00 ms

✅ 2 out of 2 shown test case(s) passed

✅ 2 out of 2 hidden test case(s) passed

✅ **Test case 1** 80 ms   🐞 Debug ≡ ▦ ∧

Expected output	Actual output
Set A: 0 2 4 5 8	Set A: 0 2 4 5 8
Set B: 1 2 3 4 5	Set B: 1 2 3 4 5
Union: {0, 1, 2, 3, 4, 5, 8}	Union: {0, 1, 2, 3, 4, 5, 8}
Intersection: {2, 4, 5}	Intersection: {2, 4, 5}
Difference: {0, 8}	Difference: {0, 8}

⯈ Terminal   ▦ Test cases

‹ Prev Reset Submit Next ›