

4.1.1] Set Operations

ALGORITHM

Step 1: Start

Step 2: Input Set A

Step 3: Convert the input values into Set A

Step 4: Input Set B

Step 5: Convert the input values into Set B

Step 6: Find the Union of Set A and Set B

Union = $A \cup B$

Step 7: Find the Intersection of Set A and Set B

Intersection = $A \cap B$

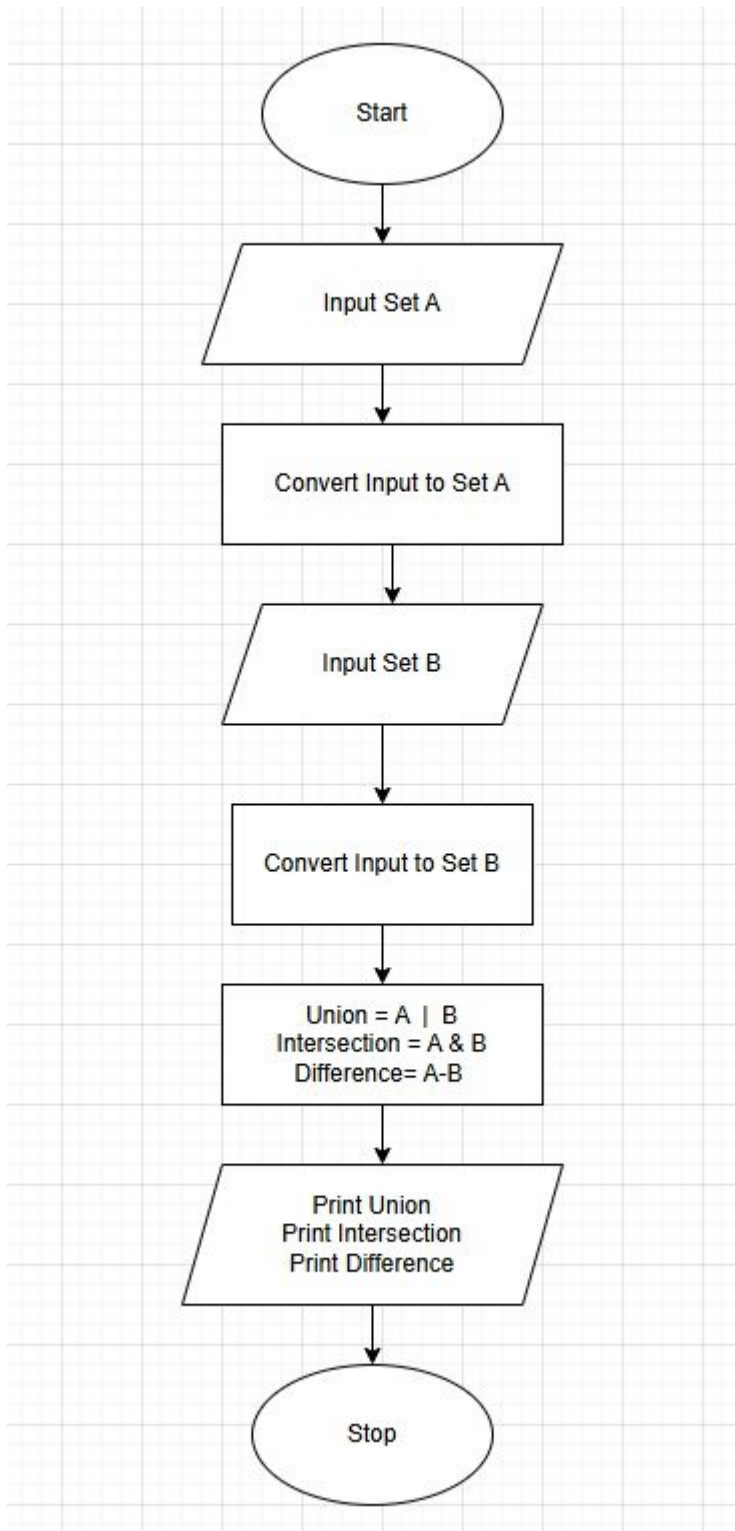
Step 8: Find the Difference of Set A and Set B

Difference = $A - B$

Step 9: Print Union, Intersection, and Difference

Step 10: Stop

FLOWCHART



PYTHON CODE

```
set_a = set(map(int, input("Set A: ").split()))
```

```
set_b = set(map(int, input("Set B: ").split()))
```

```
print("Union:", set_a | set_b)
```

```
print("Intersection:", set_a & set_b)
```

```
print("Difference:", set_a - set_b)
```

EXECUTION

The screenshot displays the CodeTANTRA IDE interface. On the left, the problem statement for "4.1.1. Set Operations" is visible, detailing the requirements for a Python program that takes two sets of integers as input and outputs their union, intersection, and difference. The input format specifies two lines of space-separated integers. The output format requires three lines of results. A note mentions that an empty set should be printed as "set()".

The main editor shows the following Python code:

```
1 # Type Content here...
2 set_a=set(map(int,input("Set A: ").split()))
3 set_b=set(map(int,input("Set B: ").split()))
4 print("Union:", set_a|set_b)
5 print("Intersection:", set_a&set_b)
6 print("Difference:", set_a-set_b)
```

On the right, the execution results are shown. The "Test case 1" status is "PASSED". The performance metrics are: Average time 0.000 s, Maximum time 0.012 s, and Memory 8.00 mb. The test case passed 2 out of 2 shown test cases and 2 out of 2 hidden test cases. The expected output is displayed as:

```
Set A: {1, 2, 3}
Set B: {2, 3, 4}
Union: {1, 2, 3, 4}
Intersection: {2, 3}
Difference: {1, 4}
```

The actual output matches the expected output. At the bottom, there are buttons for "PREV", "RESET", "SUBMIT", and "NEXT".