



Practice > Python > Sets > Symmetric Difference

Symmetric Difference ☆

69/115 challenges solved

Rank: 4093 | Points: 1415



Your Symmetric Difference submission got 10.00 points.

Share

Tweet

[Try the next challenge](#) | [Try a Random Challenge](#)

Problem

Submissions

Leaderboard

Discussions

Editorial

Objective

Today, we're learning about a new data type: *sets*.

Concept

If the inputs are given on one line separated by a space character, use *split()* to get the separate values in the form of a list:

```
>> a = raw_input()
5 4 3 2
>> lis = a.split()
>> print (lis)
['5', '4', '3', '2']
```

If the list values are all integer types, use the *map()* method to convert all the strings to integers.

```
>> newlis = list(map(int, lis))
>> print (newlis)
[5, 4, 3, 2]
```

Sets are an unordered bag of unique values. A single set contains values of any immutable data type.

CREATING SETS

```
>> myset = {1, 2} # Directly assigning values to a set
>> myset = set() # Initializing a set
>> myset = set(['a', 'b']) # Creating a set from a list
>> myset
{'a', 'b'}
```

MODIFYING SETS

Using the *add()* function:

```
>> myset.add('c')
>> myset
{'a', 'c', 'b'}
>> myset.add('a') # As 'a' already exists in the set, nothing happens
>> myset.add((5, 4))
```

Author

harsh_beria93

Difficulty

Easy

Max Score

10

Submitted By

36235

NEED HELP?

[View discussions](#)[View editorial](#)[View top submissions](#)

RATE THIS CHALLENGE



MORE DETAILS

[Download problem statement](#)[Download sample test cases](#)[Suggest Edits](#)

```
>> myset
{'a', 'c', 'b', (5, 4)}
```

Using the *update()* function:

```
>> myset.update([1, 2, 3, 4]) # update() only works for iterable objects
>> myset
{'a', 1, 'c', 'b', 4, 2, (5, 4), 3}
>> myset.update({1, 7, 8})
>> myset
{'a', 1, 'c', 'b', 4, 7, 8, 2, (5, 4), 3}
>> myset.update({1, 6}, [5, 13])
>> myset
{'a', 1, 'c', 'b', 4, 5, 6, 7, 8, 2, (5, 4), 13, 3}
```

REMOVING ITEMS

Both the *discard()* and *remove()* functions take a single value as an argument and removes that value from the set. If that value is not present, *discard()* does nothing, but *remove()* will raise a *KeyError* exception.

```
>> myset.discard(10)
>> myset
{'a', 1, 'c', 'b', 4, 5, 7, 8, 2, 12, (5, 4), 13, 11, 3}
>> myset.remove(13)
>> myset
{'a', 1, 'c', 'b', 4, 5, 7, 8, 2, 12, (5, 4), 11, 3}
```

COMMON SET OPERATIONS Using *union()*, *intersection()* and *difference()* functions.

```
>> a = {2, 4, 5, 9}
>> b = {2, 4, 11, 12}
>> a.union(b) # Values which exist in a or b
{2, 4, 5, 9, 11, 12}
>> a.intersection(b) # Values which exist in a and b
{2, 4}
>> a.difference(b) # Values which exist in a but not in b
{9, 5}
```

The *union()* and *intersection()* functions are symmetric methods:

```
>> a.union(b) == b.union(a)
True
>> a.intersection(b) == b.intersection(a)
True
>> a.difference(b) == b.difference(a)
False
```

These [other built-in data structures in Python](#) are also useful.

Task

Given **2** sets of integers, ***M*** and ***N***, print their symmetric difference in ascending order. The term *symmetric difference* indicates those values that exist in either ***M*** or ***N*** but do not exist in both.

Input Format

The first line of input contains an integer, M .

The second line contains M space-separated integers.

The third line contains an integer, N .

The fourth line contains N space-separated integers.

Output Format

Output the symmetric difference integers in ascending order, one per line.

Sample Input

```
4
2 4 5 9
4
2 4 11 12
```

Sample Output

```
5
9
11
12
```

Current Buffer (saved locally, editable)



Python 3



```
1 n=int(input())
2 a=set(input().split())
3 m=int(input())
4 b=set(input().split())
5 c=[int(i) for i in a.symmetric_difference(b)]
6 [print(i) for i in sorted(c)]
7
```

Line: 6 Col: 28

Upload Code as File

☐ Test against custom input

Run Code

Submit Code



You have earned 10.00 points!

69/115 challenges solved.

60%

Congratulations

Next
Challenge

You solved this challenge. Would you like to challenge your friends?

Challenge

- ✔ Testcase 0
- ✔ Testcase 1
- ✔ Testcase 2
- ✔ Testcase 3

10 Testcases

Input (stdin)	Download	Expected Output	Download
4		5	
2 4 5 9		9	
4		11	
2 4 11 12		12	

Compiler Message

Success