

# GE19211 / GE23233 / GE23231 – PSPP/PUP

Dashboard / My courses / PSPP/PUP / Experiments based on Tuples, Sets and its operations / Week7\_Coding

## Quiz navigation

1	2	3	4	5
✓	✓	✓	✓	✓

Show one page at a time

Finish review

Started on	Friday, 24 May 2024, 8:52 AM
State	Finished
Completed on	Friday, 24 May 2024, 8:59 AM
Time taken	6 mins 48 secs
Marks	5.00/5.00
Grade	100.00 out of 100.00

Question **1**  
Correct  
Mark 1.00 out of 1.00  
Flag question

There is a malfunctioning keyboard where some letter keys do not work. All other keys on the keyboard work properly.

Given a string text of words separated by a single space (no leading or trailing spaces) and a string brokenLetters of all distinct letter keys that are broken, return the number of words in text you can fully type using this keyboard.

Example 1:

Input: text = "hello world", brokenLetters = "ad"

Output:

1

Explanation: We cannot type "world" because the 'd' key is broken.

For example:

Input	Result
hello world ad	1
Faculty Upskilling in Python Programming ak	2

Answer: (penalty regime: 0 %)

```
1 a=input()
2 b=input()
3 c=[]
4 for char in a:
5     if char in b and char not in c:
6         c.append(char)
7 results="",join(c)
8 res=len(c)
9 print(res)
```

	Input	Expected	Got	
✓	hello world ad	1	1	✓
✓	Welcome to REC e	1	1	✓
✓	Faculty Upskilling in Python Programming ak	2	2	✓

Passed all tests! ✓

Correct  
Marks for this submission: 1.00/1.00.

Question **2**  
Correct  
Mark 1.00 out of 1.00  
Flag question

Write a program to eliminate the common elements in the given 2 arrays and print only the non-repeating elements and the total number of such non-repeating elements.

Input Format:

The first line contains space-separated values, denoting the size of the two arrays in integer format respectively.

The next two lines contain the space-separated integer arrays to be compared.

Sample Input:

5 4

1 2 8 6 5

2 6 8 10

Sample Output:

1 5 10

3

Sample Input:

5 5

1 2 3 4 5

1 2 3 4 5

Sample Output:

NO SUCH ELEMENTS

For example:

Input	Result
5 4 1 2 8 6 5 2 6 8 10	1 5 10 3

Answer: (penalty regime: 0 %)

```
1 n, m = map(int, input().split())
2 array1 = list(map(int, input().split()))
3 array2 = list(map(int, input().split()))
4 set1 = set(array1)
5 set2 = set(array2)
6 symmetric_diff = set1.symmetric_difference(set2)
7 non_repeating_elements = [x for x in symmetric_diff if x not in set1 or x not in set2]
8 if non_repeating_elements:
9     print("non_repeating_elements")
10    print(len(non_repeating_elements))
11 else:
12    print("NO SUCH ELEMENTS")
```

	Input	Expected	Got	
✓	5 4 1 2 8 6 5 2 6 8 10	1 5 10 3	1 5 10 3	✓
✓	3 3 10 10 10 10 11 12	11 12 2	11 12 2	✓

Passed all tests! ✓

Correct  
Marks for this submission: 1.00/1.00.

Question **3**  
Correct  
Mark 1.00 out of 1.00  
Flag question

The **DNA sequence** is composed of a series of nucleotides abbreviated as 'A', 'C', 'G', and 'T'.

- For example, "ACGAATTCGC" is a **DNA sequence**.

When studying **DNA**, it is useful to identify repeated sequences within the DNA.

Given a string s that represents a **DNA sequence**, return all the **10-letter-long** sequences (substrings) that occur more than once in a DNA molecule. You may return the answer in **any order**.

Example 1:

Input: s = "AAAAACCCCCAAAAACCCCCCAAAAGGGTTT"

Output: ["AAAAACCCCC", "CCCCCAAAAA"]

Example 2:

Input: s = "AAAAAAAAAAAA"

Output: ["AAAAAAAAAA"]

For example:

Input	Result
AAAAACCCCCAAAAACCCCCCAAAAGGGTTT	AAAAACCCCC CCCCCAAAAA

Answer: (penalty regime: 0 %)

```
1 s = input()
2 A = set()
3 B = set()
4 for i in range(len(s) - 9):
5     C = s[i:i + 10]
6     if C in A:
7         B.add(C)
8     else:
9         A.add(C)
10    for seq in B:
11        print(seq)
```

	Input	Expected	Got	
✓	AAAAACCCCCAAAAACCCCCCAAAAGGGTTT	AAAAACCCCC CCCCCAAAAA	AAAAACCCCC CCCCCAAAAA	✓
✓	AAAAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	✓

Passed all tests! ✓

Correct  
Marks for this submission: 1.00/1.00.

Question **4**  
Correct  
Mark 1.00 out of 1.00  
Flag question

Coders here is a simple task for you, Given string str. Your task is to check whether it is a binary string or not by using python set.

Examples:

Input: str = "01010101010"

Output: Yes

Input: str = "REC101"

Output: No

For example:

Input	Result
01010101010	Yes
010101 10101	No

Answer: (penalty regime: 0 %)

```
1 def is_binary_string(s):
2     binary_set = {'0', '1'}
3     return set(s).issubset(binary_set)
4
5 def main():
6     s = input().strip()
7     if is_binary_string(s):
8         print("Yes")
9     else:
10        print("No")
11
12 if __name__ == "__main__":
13     main()
14
```

	Input	Expected	Got	
✓	01010101010	Yes	Yes	✓
✓	REC123	No	No	✓
✓	010101 10101	No	No	✓

Passed all tests! ✓

Correct  
Marks for this submission: 1.00/1.00.

Question **5**  
Correct  
Mark 1.00 out of 1.00  
Flag question

Given a tuple and a positive integer k, the task is to find the count of distinct pairs in the tuple whose sum is equal to **K**.

Examples:

Input: t = (5, 6, 5, 7, 7, 8), K = 13

Output: 2

Explanation:

Pairs with sum K( = 13) are {(5, 8), (6, 7), (6, 7)}.

Therefore, distinct pairs with sum K( = 13) are { (5, 8), (6, 7) }.

Therefore, the required output is 2.

For example:

Input	Result
1,2,1,2,5 3	1
1,2 0	0

Answer: (penalty regime: 0 %)

```
1 t = tuple(map(int, input().split(',')))
2 K = int(input())
3 pairs = set()
4 for i in range(len(t)):
5     for j in range(i + 1, len(t)):
6         if t[i] + t[j] == K:
7             pairs.add((min(t[i], t[j]), max(t[i], t[j])))
8 print(len(pairs))
```

	Input	Expected	Got	
✓	5,6,5,7,7,8 13	2	2	✓
✓	1,2,1,2,5 3	1	1	✓
✓	1,2 0	0	0	✓

Passed all tests! ✓

Correct  
Marks for this submission: 1.00/1.00.

Finish review

◀ Week7\_MCQ

Jump to...

Dictionary ▶

Contact site support

You are logged in as ASWATHY K B 2022-BIOMED-A (Log out)

PSPP/PUP

Data retention summary