

Week 2

Week 2 of Programming for Data Scientists: Python



Problems

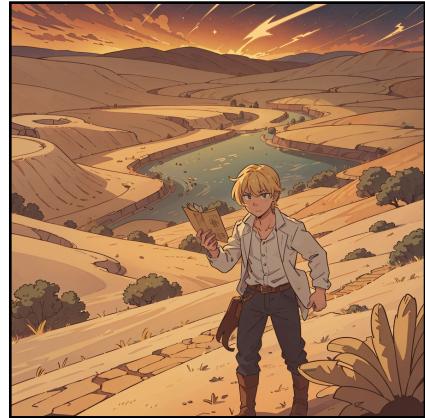
- 02-A Wild West Wordplay
- 02-B Numerical Nuggets
- 02-C Palindrome Prowess
- 02-D Secrets in the Sands
- 02-E Daring Duel

02-A Wild West Wordplay

Time limit: 10s

In the scorching heat of the Wild West, where tumbleweeds rolled across vast deserts and gold mines glittered in the distance, there lived a peculiar cowboy named Slim. Slim was not your ordinary cowboy; he was known far and wide for his extraordinary ability to manipulate strings.

Legend has it that Slim stumbled upon an ancient treasure map while he was out chasing outlaws. The map, written in cryptic symbols, hinted at a secret gold mine hidden deep within the desert. To decipher the map, Slim needed to extract specific characters from strings scattered around the Wild West.



Your task is to help Slim unlock the secret by writing a program that performs a special kind of string concatenation. Here's what it needs to do:

1. Take a string as input.
2. If the string has less than 2 characters, the program should say 0.
3. If the string has 2 or more characters, the program should make a new string. This string is made by joining the first 2 and last 2 characters of the input string.
4. Then, the program should show the new string.

Input

The input consists of a single non-empty string, provided via the standard input. The string will contain only common characters.

Output

Print the resultant string based on the criteria mentioned above.

Sample Input 1

Wild West	
-----------	--

Sample Output 1

	Wist
--	------

Sample Input 2

X	0
---	---

Sample Output 2

	0
--	---

02-B Numerical Nuggets

Time limit: 10s

In the heart of the Wild West, where dust storms swirled around ghost towns and the echoes of gunshots filled the air, Slim the string-slinging cowboy had successfully deciphered the first clue of the ancient treasure map. But the map is not yet complete. To unravel the final mystery and reach the hidden gold mine, Slim needs to master the art of number sequencing.

Legend whispers of arithmetic sequences inscribed on the walls of an old saloon. The sequence is said to hold the key to finding the hidden gold mines location. To harness this power, Slim has to carefully select odd numbers from one sequence and even numbers from another, combining them into a new sequence.

Your task is to help Slim construct the ultimate sequence that will lead him to the legendary treasure. Here are the rules:

1. Construct a new sequence using all the odd integers from the first sequence and all the even integers from the second sequence.
2. Sort the new sequence in descending order.
3. Output the new sequence of integers as a list, the key to deciphering the final clue of the treasure map.



Input

The input consists of two lines. Each line represents a sequence of integers separated by spaces.¹ Each sequence will have at least one integer.

Output

Output the new sequence of integers as a list. Remember to sort the sequence in descending order.

Sample Input 1

1 2 3 4	[8, 6, 3, 1]
5 6 7 8	

Sample Output 1

Sample Input 2

1 2 3	[3, 2, 1]
1 2 3	

Sample Output 2

¹Remember 01-C if you are struggling with reading the input.

02-C Palindrome Prowess

Time limit: 10s

The town's annual rodeo was fast approaching, and the organizers had decided to incorporate palindromes into the event. They believed that finding palindromes would bring good luck to the rodeo participants.

Slim, being the wordsmith that he was, took on the challenge. He needed to inspect various words provided by the rodeo organizers and determine if they were palindromes. It wasn't just about the rodeo; his reputation was at stake, and Slim was determined to prove to the Wild West that he had a way with words.



Given a string, Slim has to determine if it is a palindrome.

Similar to the other townsfolk, the organizers only care about the alphanumeric characters in the string and ignore the case of the characters. A palindrome is a word, phrase, number, or other sequence of characters that reads the same forward and backward.

Input

A single line containing a string s . The string may contain any characters, including spaces and punctuation marks. Ignore all punctuation and capitalization.

Output

Output `True` if the string s is a palindrome after ignoring the case of the characters.² Output `False` otherwise.

Sample Input 1

Sample Output 1

Bob	True
-----	------

Sample Input 2

Sample Output 2

Wild West	False
-----------	-------

Input 3

A man, a plan, a canal, Panama!

Output 3

True

²For example, Sample 3 should be considered a palindrome, though there are spaces and punctuation marks.

02-D Secrets in the Sands

Time limit: 10s

In the busy town of Dusty Trails, where secrets were as abundant as the golden sands in the desert, the need for secret messages held paramount importance. Slim, having settled in the town after now discovering the hidden gold mine, found himself deeply involved in Dusty Trails' complex society.

To safeguard confidentiality, the townsfolk employed a system involving the sender's message and a reference sequence.

Slim knew that if the same words appeared in both the sender's message and the reference sequence, it could lead to confusion and misinterpretation. A misplaced word or a misread message could have dire consequences, especially in the Wild West where alliances were fragile, and trust was hard-earned.

Hence, the people of Dusty Trails turned to Slim to devise a reliable method to guarantee the uniqueness of their messages. Slim now has to review the words provided by the sender, compare them with the reference sequence, and determine whether each word is truly unique, in order to prevent misunderstandings, uphold trust, and maintain the delicate balance of Dusty Trails' social dynamics.

Another observation Slim has made while living in town is that the locals don't place much importance on punctuation or capitalization. Therefore, you should disregard them. He doesn't want to be bothered by these trivialities.



Input

The input consists of two lines:

- The first line contains strings separated by spaces, representing the sender's message.
- The second line contains strings separated by spaces, representing the reference sequence.

Ignore all punctuation and capitalization.

Output

Print `True` if all strings in the sender's message are unique with respect to the reference sequence. Here, string is defined as a sequence of characters separated by spaces. Be careful with capitalization and punctuation. Have a look at the `punctuation` string from the standard library, which should help you catch all punctuation characters. Print `False` otherwise. This should help the townsfolk avoid confusion and misunderstandings.

Input 1

I live in the town of Dusty Trails.
This is my favourite town!

Output 1

False

Input 2

At the stroke of midnight the gold glimmers under the old oak.
As dawn approaches, townspeople gather for a grand reveal...

Output 2

True

Sample Input 3

I want to find some gold.
WE FOUND GOLD!

Sample Output 3

False

Sample Input 4

dusty-trails
dusty'trails

Sample Output 4

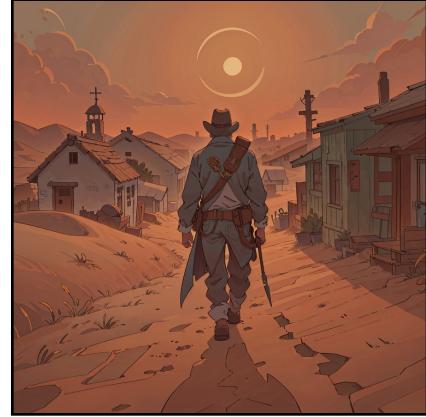
False

02-E Daring Duel

Time limit: 10s

Slim's sharp wit and keen mind had made him a famous duelist amongst cowboys in the Wild West. In Dusty Trails, where duels are not fought with pistols but with intellect, Slim's abilities quickly gained him the people's trust, so they soon started relying on him as the town's protector.

One day, a mysterious outsider known only as The Enigma poses an intriguing challenge to Slim. The Enigma, renowned for their mathematical prowess and love for riddles, leaves behind a perplexing fraction encoded in various ways, from scientific notation to cleverly arranged spaces. This enigmatic fraction is rumored to unlock a power never seen before, turning anyone who deciphered it into an unbeatable duelist. Intrigued and undeterred, Slim eagerly accepts the challenge, ready to uncover the hidden mathematical treasures within the fraction.



You are given a string representation of a fraction, possibly in various forms including scientific notation and may contain whitespace characters. Your task is to use Python's *fractions* module to handle this string and output its simplified form.

- Read about the `fractions` module from the [Python Fractions Documentation](#).
- Import the `fractions` module into your code.
- Parse the given string to create a `Fraction` object.
- Print the following, each on a new line:
 - The numerator of the fraction.
 - The denominator of the fraction.
 - The entire fraction in its simplest form. (Take a look at the methods of the `Fraction` class)

Input

A single line containing a string representation of a fraction. This string can be in various forms, including but not limited to scientific notation.

Output

Three lines of output:

- The numerator of the fraction.

- The denominator of the fraction.
- The entire fraction in its simplest form.³

Sample Input 1	Sample Output 1
1/2	1 2 1/2
Sample Input 2	Sample Output 2
0 . 8	4 5 4/5
Sample Input 3	Sample Output 3
5e-2	1 20 1/20
Sample Input 4	Sample Output 4
2 / 3	2 3 2/3

³The most basic form of a fraction occurs when the numerator and denominator have no common divisors aside from 1. In cases where the fraction is negative, the negative sign should appear on the numerator. If the denominator equals 1, the fraction should be represented as an integer. This conversion is automated by the mentioned module.