# Python Programming Puzzles

**1.** Write a Python program find a list of integers with exactly two occurrences of nineteen and at least three occurrences of five.
Input:
[19, 19, 15, 5, 3, 5, 5, 2]
Output:
True
Input:
[19, 15, 15, 5, 3, 3, 5, 2]
Output:
False
Input:
[19, 19, 5, 5, 5, 5, 5]
Output:
True

**2.** Write a Python program that accept a list of integers and check the length and the fifth element. Return true if the length of the list is 8 and fifth element occurs thrice in the said list.
Input:
[19, 19, 15, 5, 5, 5, 1, 2]
Output:
True
Input:
[19, 15, 5, 7, 5, 5, 2]
Output:
False
Input:
[11, 12, 14, 13, 14, 13, 15, 14]
Output:
True
Input:
[19, 15, 11, 7, 5, 6, 2]
Output:
False

**3.** Write a Python program that accept an integer test whether an integer greater than 4^4 and which is 4 mod 34.
Input:
922
Output:
True
Input:
914
Output:
False
Input:
854
Output:
True
Input:
854
Output:
True

**4.** We are making n stone piles! The first pile has n stones. If n is even, then all piles have an even number of stones. If n is odd, all piles have an odd number of stones. Each pile must more stones than the previous pile but as few as possible. Write a Python program to find the number of stones in each pile.
Input: 2
Output:
[2, 4]
Input: 10
Output:
[10, 12, 14, 16, 18, 20, 22, 24, 26, 28]
Input: 3
Output:
[3, 5, 7]
Input: 17
Output:
[17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49]

**5.** Write a Python program to check the $n^{th-1}$ string is a proper substring of $n^{th}$ string in a given list of strings.
Input:
['a', 'abb', 'sfs', 'oo', 'de', 'sfde']
Output:
True
Input:
['a', 'abb', 'sfs', 'oo', 'ee', 'sfde']
Output:
False
Input:
['a', 'abb', 'sad', 'ooaaesdfe', 'sfsdfde', 'sfsd', 'sfsdf', 'qwrew']
Output:
False
Input:
['a', 'abb', 'sad', 'ooaaesdfe', 'sfsdfde', 'sfsd', 'sfsdf', 'qwsfsdfrew']
Output:
True

**6.** Write a Python program to test a list of one hundred integers between 0 and 999, which all differ by ten from one another. Return true or false.
Input:
[0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520, 530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650, 660, 670, 680, 690, 700, 710, 720, 730, 740, 750, 760, 770, 780, 790, 800, 810, 820, 830, 840, 850, 860, 870, 880, 890, 900, 910, 920, 930, 940, 950, 960, 970, 980, 990]
Output:
True
Input:
[0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 280, 300, 320, 340, 360, 380, 400, 420, 440, 460, 480, 500, 520, 540, 560, 580, 600, 620, 640, 660, 680, 700, 720, 740, 760, 780, 800, 820, 840, 860, 880, 900, 920, 940, 960, 980]
Output:
False

**7.** Write a Python program to check a given list of integers where the sum of the first i integers is i.
Input:
[0, 1, 2, 3, 4, 5]
Output:
False
Input:
[1, 1, 1, 1, 1, 1]
Output:
True
Input:
[2, 2, 2, 2, 2]
Output:
False

**8.** Write a Python program to split a string of words separated by commas and spaces into two lists, words and separators.
Input: W3resource Python, Exercises.
Output:
[['W3resource', 'Python', 'Exercises.'], [' ', ', ']]
Input: The dance, held in the school gym, ended at midnight.
Output:
[['The', 'dance', 'held', 'in', 'the', 'school', 'gym', 'ended', 'at', 'midnight.'], [' ', ', ', ' ', ' ', ' ', ' ', ', ', ' ', ' ']]
Input: The colors in my studyroom are blue, green, and yellow.
Output:
[['The', 'colors', 'in', 'my', 'studyroom', 'are', 'blue', 'green', 'and', 'yellow.'], [' ', ' ', ' ', ' ', ' ', ' ', ', ', ', ', ', ' ']]

**9.** Write a Python program to find list integers containing exactly four distinct values, such that no integer repeats twice consecutively among the first twenty entries.
Input:
[1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4]
Output:
True
Input:
[1, 2, 3, 3, 1, 2, 3, 3, 1, 2, 3, 3, 1, 2, 3, 3]
Output:
False
Input:
[1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3]
Output:
False

**10.** Given a string consisting of whitespace and groups of matched parentheses, write a Python program to split it into groups of perfectly matched parentheses without any whitespace.
Input:
( ()) (((()()())) (()) ()
Output:
['(())', '(((()()()))', '(())', '()']
Input:
() (( ( )() ( )) ) ( ())
Output:
['()', '(((()()()))', '(())']

**11.** Write a Python program to find the indexes of numbers of a given list below a given threshold.
Original list:
[0, 12, 45, 3, 4923, 322, 105, 29, 15, 39, 55]
Threshold: 100
Check the indexes of numbers of the said list below the given threshold:
[0, 1, 2, 3, 7, 8, 9, 10]
Original list:
[0, 12, 4, 3, 49, 9, 1, 5, 3]
Threshold: 10
Check the indexes of numbers of the said list below the given threshold:
[0, 2, 3, 5, 6, 7, 8]

**12.** Write a Python program to check whether the given strings are palindromes or not. Return True, False.
Input:
['palindrome', 'madamimadam', '', 'foo', 'eyes']
Output:
[False, True, True, False, False]

**13.** Write a Python program to find the strings in a given list, starting with a given prefix.
Input:
[( ca,('cat', 'car', 'fear', 'center'))]
Output:
['cat', 'car']
Input:
[(do,('cat', 'dog', 'shatter', 'donut', 'at', 'todo'))]
Output:
['dog', 'donut']

**14.** Write a Python program to find the lengths of a given list of non-empty strings.
Input:
['cat', 'car', 'fear', 'center']
Output:
[3, 3, 4, 6]
Input:
['cat', 'dog', 'shatter', 'donut', 'at', 'todo', '']
Output:
[3, 3, 7, 5, 2, 4, 0]

**15.** Write a Python program find the longest string of a given list of strings.
Input:
['cat', 'car', 'fear', 'center']
Output:
center
Input:
['cat', 'dog', 'shatter', 'donut', 'at', 'todo', '']
Output:
shatter

**16.** Write a Python program find the strings in a given list containing a given substring.
Input:
[(ca,('cat', 'car', 'fear', 'center'))]
Output:
['cat', 'car']
Input:
[(o,('cat', 'dog', 'shatter', 'donut', 'at', 'todo', ''))]
Output:
['dog', 'donut', 'todo']
Input:
[(oe,('cat', 'dog', 'shatter', 'donut', 'at', 'todo', ''))]
Output:
[]

**17.** Write a Python program to create string consisting of the non-negative integers up to n inclusive.
Input:
4
Output:
0 1 2 3 4
Input:
15
Output:
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

**18.** An irregular/uneven matrix, or ragged matrix, is a matrix that has a different number of elements in each row. Ragged matrices are not used in linear algebra, since standard matrix transformations cannot be performed on them, but they are useful as arrays in computing.
Write a Python program to find the indices of all occurrences of target in the uneven matrix.
Input:
[([1, 3, 2, 32, 19], [19, 2, 48, 19], [], [9, 35, 4], [3, 19]),19]
Output:
[[0, 4], [1, 0], [1, 3], [4, 1]]
Input:
[([1, 2, 3, 2], [], [7, 9, 2, 1, 4]),2]
Output:
[[0, 1], [0, 3], [2, 2]]

**19.** Write a Python program to split a given string (s) into strings if there is a space in the string, otherwise split on commas if there is a comma, otherwise return the list of lowercase letters with odd order (order of a = 0, b = 1, etc.)
Input:
a b c d
Split the said string into strings if there is a space in the string,
otherwise split on commas if there is a comma,
Output:
['a', 'b', 'c', 'd']
Input:
a,b,c,d
Split the said string into strings if there is a space in the string,
otherwise split on commas if there is a comma,
Output:
['a', 'b', 'c', 'd']
Input:
abcd
Split the said string into strings if there is a space in the string,
otherwise split on commas if there is a comma,
Output:
['b', 'd']

**20.** Write a Python program to determine the direction ('increasing' or 'decreasing') of monotonic sequence numbers.
Input:
[1, 2, 3, 4, 5, 6]
Output:
Increasing.
Input:
[6, 5, 4, 3, 2, 1]
Output:
Decreasing.
Input:
[19, 19, 5, 5, 5, 5, 5]
Output:
Not a monotonic sequence!

**21.** Write a Python program to check, for each string in a given list, whether the last character is an isolated letter or not. Return True or False.
Input:
['cat', 'car', 'fear', 'center']
Output:
[False, False, False, False]
Input:
['ca t', 'car', 'fea r', 'cente r']
Output:
[True, False, True, True]

**22.** Write a Python program to compute the sum of the ASCII values of the upper-case characters in a given string.
Input:
PytHon ExerciSEs
Output:
373
Input:
JavaScript
Output:
157

**23.** Write a Python program to find the indices for which the numbers in the list drops.
NOTE: You can detect multiple drops just by checking if nums[i] < nums[i-1]
Input:
[0, -1, 3, 8, 5, 9, 8, 14, 2, 4, 3, -10, 10, 17, 41, 22, -4, -4, -15, 0]
Output:
[1, 4, 6, 8, 10, 11, 15, 16, 18]
Input:
[6, 5, 4, 3, 2, 1]
Output:
[1, 2, 3, 4, 5]
Input:
[1, 19, 5, 15, 5, 25, 5]
Output:
[0, 2, 4, 6]

**24.** Write a Python program to create a list whose $i^{th}$ element is the maximum of the first i elements from a input list.
Input:
[0, -1, 3, 8, 5, 9, 8, 14, 2, 4, 3, -10, 10, 17, 41, 22, -4, -4, -15, 0]
Output:
[0, 0, 3, 8, 8, 9, 9, 14, 14, 14, 14, 14, 14, 17, 41, 41, 41, 41, 41, 41]
Input:
[6, 5, 4, 3, 2, 1]
Output:
[6, 6, 6, 6, 6, 6]
Input:
[1, 19, 5, 15, 5, 25, 5]
Output:
[1, 19, 19, 19, 19, 25, 25]

**25.** Write a Python program to find the XOR of two given strings interpreted as binary numbers.
Input:
['0001', '1011']
Output:
0b1010
Input:
['100011101100001', '100101100101110']
Output:
0b110001001111

**26.** Write a Python program to find the largest number where commas or periods are decimal points.
Input:
['100', '102,1', '101.1']
Output:
102.1

**27.** Write a Python program to find x that minimizes mean squared deviation from a given a list of numbers.
Input:
[4, -5, 17, -9, 14, 108, -9]
Output:
17.142857142857142
Input:
[12, -2, 14, 3, -15, 10, -45, 3, 30]
Output:
1.1111111111111112

**28.** Write a Python program to select a string from a given list of strings with the most unique characters.
Input:
['cat', 'catatatatctsa', 'abcdefhijklmnop', '124259239185125', '', 'foo', 'unique']
Output:
abcdefhijklmnop
Input:
['Green', 'Red', 'Orange', 'Yellow', '', 'White']
Output:
Orange

**29.** Write a Python program to find the indices of two numbers that sum to 0 in a given list of numbers.
Input:
[1, -4, 6, 7, 4]
Output:
[4, 1]
Input:
[1232, -20352, 12547, 12440, 741, 341, 525, 20352, 91, 20]
Output:
[1, 7]

**30.** Write a Python program to find the list of strings that has fewer total characters (including repetitions).
Input:
[['this', 'list', 'is', 'narrow'], ['I', 'am', 'shorter but wider']]
Output:
['this', 'list', 'is', 'narrow']
Input:
[['Red', 'Black', 'Pink'], ['Green', 'Red', 'White']]
Output:
['Red', 'Black', 'Pink']

**31.** Write a Python program to find the coordinates of a triangle with the given side lengths.
Input:
[3, 4, 5]
Output:
[[0.0, 0.0], [3, 0.0], [3.0, 4.0]]
Input:
[5, 6, 7]
Output:
[[0.0, 0.0], [5, 0.0], [3.8, 5.878775382679628]]

**32.** Write a Python program to rescale and shift numbers of a given list, so that they cover the range [0, 1].
Input:
[18.5, 17.0, 18.0, 19.0, 18.0]
Output:
[0.75, 0.0, 0.5, 1.0, 0.5]
Input:
[13.0, 17.0, 17.0, 15.5, 2.94]
Output:
[0.7155049786628734, 1.0, 1.0, 0.8933143669985776, 0.0]

**33.** Write a Python program to find the positions of all uppercase vowels (not counting Y) in even indices of a given string.
Input: w3rEsOUrcE
Output:
[6]
Input: AEIOUYW
Output:
[0, 2, 4]

**34.** Write a Python program to find the sum of the numbers of a given list among the first k with more than 2 digits.
Input: [4, 5, 17, 9, 14, 108, -9, 12, 76]
Value of K: 4
Output:
0
Input: [4, 5, 17, 9, 14, 108, -9, 12, 76]
Value of K: 6
Output:
108
Input: [114, 215, -117, 119, 14, 108, -9, 12, 76]
Value of K: 5
Output:
331
Input: [114, 215, -117, 119, 14, 108, -9, 12, 76]
Value of K: 1
Output:
114

**35.** Write a Python program to compute the product of the odd digits in a given number, or 0 if there aren't any.
Input: 123456789
Output:
945
Input: 2468
Output:
0
Input: 13579
Output:
945

**36.** Write a Python program to find the largest k numbers from a given list of numbers.
Input: [1, 2, 3, 4, 5, 5, 3, 6, 2]
Output:
[6]
Input: [1, 2, 3, 4, 5, 5, 3, 6, 2]
Output:
[6, 5]
Input: [1, 2, 3, 4, 5, 5, 3, 6, 2]
Output:
[6, 5, 5]
Input: [1, 2, 3, 4, 5, 5, 3, 6, 2]
Output:
[6, 5, 5, 4]
Input: [1, 2, 3, 4, 5, 5, 3, 6, 2]
Output: [6, 5, 5, 4, 3]

**37.** Write a Python program to find the largest integer divisor of a number n that is less than n.
Input: 18
Output:
9
Input: 100
Output:
50
Input: 102
Output:
51
Input: 500
Output:
250
Input: 1000
Output:
500
Input: 6500
Output:
3250

**38.** Write a Python program to sort the numbers of a given list by the sum of their digits.
Input: [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
Output:
[10, 11, 20, 12, 13, 14, 15, 16, 17, 18, 19]
Input: [23, 2, 9, 34, 8, 9, 10, 74]
Output:
[10, 2, 23, 34, 8, 9, 9, 74]

**39.** Write a Python program to determine which triples sum to zero from a given list of lists.
Input: [[1343532, -2920635, 332], [-27, 18, 9], [4, 0, -4], [2, 2, 2], [-20, 16, 4]]
Output:
[False, True, True, False, True]
Input: [[1, 2, -3], [-4, 0, 4], [0, 1, -5], [1, 1, 1], [-2, 4, -1]]
Output:
[True, True, False, False, False]

**40.** Write a Python program to find string s that, when case is flipped gives target where vowels are replaced by chars two later.
Input: Python
Output:
pYTHQN
Input: aeiou
Output:
CGKQW
Input: Hello, world!
Output:
hGLLQ, WQRLD!
Input: AEIOU
Output:
cgkqw

**41.** Write a Python program to sort numbers based on strings.
Input: six one four one two three
Output:
one two three four six
Input: six one four three two nine eight
Output:
one two three four six eight nine
Input: nine eight seven six five four three two one
Output:
one two three four five six seven eight nine

**42.** Write a Python program to find the set of distinct characters in a given string, ignoring case.
Input: HELLO
Output:
['h', 'o', 'l', 'e']
Input: HelLo
Output:
['h', 'o', 'l', 'e']
Input: Ignoring case
Output:
['s', 'n', 'c', 'o', 'e', 'i', 'r', 'g', 'a', ' ']

**43.** Write a Python program to find all words in a given string with n consonants.
Input: this is our time
Output:
Number of consonants: 3
Words in the said string with 3 consonants:
['this']
Number of consonants: 2
Words in the said string with 2 consonants:
['time']
Number of consonants: 1
Words in the said string with 1 consonants:
['is', 'our']

**44.** Write a Python program to find which characters of a hexadecimal number correspond to prime numbers.
Input: 123ABCD
Output:
[False, True, True, False, True, False, True]
Input: 123456
Output:
[False, True, True, False, True, False]
Input: FACE
Output:
[False, False, False, False]

**45.** Write a Python program to find all even palindromes up to n.
Output:
Even palindromes up to 50 -
[0, 2, 4, 6, 8, 22, 44]
Even palindromes up to 100 -
[0, 2, 4, 6, 8, 22, 44, 66, 88]
Even palindromes up to 500 -
[0, 2, 4, 6, 8, 22, 44, 66, 88, 202, 212, 222, 232, 242, 252, 262, 272, 282, 292, 404, 414, 424, 434, 444, 454, 464, 474, 484, 494]
Even palindromes up to 2000 -
[0, 2, 4, 6, 8, 22, 44, 66, 88, 202, 212, 222, 232, 242, 252, 262, 272, 282, 292, 404, 414, 424, 434, 444, 454, 464, 474, 484, 494, 606, 616, 626, 636, 646, 656, 666, 676, 686, 696, 808, 818, 828, 838, 848, 858, 868, 878, 888, 898]

**46.** Given an array of numbers representing a branch on a binary tree, write a Python program to find the minimum even value and its index. In the case of a tie, return the smallest index. If there are no even numbers, the answer is [].
Input:
[1, 9, 4, 6, 10, 11, 14, 8]
Output:
Minimum even value and its index of the said array of numbers:
[4, 2]
Input:
[1, 7, 4, 4, 9, 2]
Output:
Minimum even value and its index of the said array of numbers:
[2, 5]
Input:
[1, 7, 7, 5, 9]
Output:
Minimum even value and its index of the said array of numbers:
[]

**47.** Write a Python program to Filter for the numbers in numbers in a given list whose sum of digits is > 0, where the first digit can be negative.
Input:
[11, -6, -103, -200]
Output:
[11, -103]
Input:
[1, 7, -4, 4, -9, 2]
Output:
[1, 7, 4, 2]
Input:
[10, -11, -71, -13, 14, -32]
Output:
[10, -13, 14]

**48.** Write a Python program to find the indices of two entries that show that the list is not in increasing order. If there are no violations (they are increasing), return an empty list.
Input: [1, 2, 3, 0, 4, 5, 6]
Output: [2, 3]
Input: [1, 2, 3, 4, 5, 6]
Output: []
Input: [1, 2, 3, 4, 6, 5, 7]
Output: [4, 5]
Input: [-3, -2, -3, 0, 2, 3, 4]
Output: [1, 2]

**49.** Write a Python program to find the h-index, the largest positive number h such that h occurs in the sequence at least h times. If there is no such positive number return h = -1.
Input:
[1, 2, 2, 3, 3, 4, 4, 4, 4]
Output:
4
Input:
[1, 2, 2, 3, 4, 5, 6]
Output:
2
Input:
[3, 1, 4, 17, 5, 17, 2, 1, 41, 32, 2, 5, 5, 5, 5]
Output:
5

**50.** Write a Python program to find the even-length words from a given list of words and sort them by length.
Original list of words:
['Red', 'Black', 'White', 'Green', 'Pink', 'Orange']
Find the even-length words and sort them by length in the said list of words:
['Pink', 'Orange']
Original list of words:
['The', 'worm', 'ate', 'a', 'bird', 'imagine', 'that', '!', 'Absurd', '!!']
Find the even-length words and sort them by length in the said list of words:
['!!', 'bird', 'that', 'worm', 'Absurd']

**51.** Write a Python program to find the first n Fibonacci numbers.
Input: 10
Output:
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
Input: 15
Output:
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610]
Input: 50
Output:
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811, 514229, 832040, 1346269, 2178309, 3524578, 5702887, 9227465, 14930352, 24157817, 39088169, 63245986, 102334155, 165580141, 267914296, 433494437, 701408733, 1134903170, 1836311903, 2971215073, 4807526976, 7778742049, 12586269025]

**52.** Write a Python program to reverse the case of all strings. For those strings, which contain no letters, reverse the strings.
Original list:
['cat', 'catatatatctsa', 'abcdefhijklmnop', '124259239185125', '', 'foo', 'unique']
Reverse the case of all strings. For those strings which contain no letters, reverse the strings:
['CAT', 'CATATATATCTSA', 'ABCDEFHIJKLMNOP', '521581932952421', '', 'FOO', 'UNIQUE']
Original list:
['Green', 'Red', 'Orange', 'Yellow', '', 'White']
Reverse the case of all strings. For those strings which contain no letters, reverse the strings:
['gREEN', 'rED', 'oRANGE', 'yELLOW', '', 'wHITE']
Original list:
['Hello', '!@#', '!@#$', '123#@!']
Reverse the case of all strings. For those strings which contain no letters, reverse the strings:
['hELLO', '#@!', '$#@!', '!@#321']

**53.** Write a Python program to find the product of the units digits in the numbers of a given list.
Input:
[12, 23]
Output:
6
Input:
[12, 23, 43]
Output:
18
Input:
[113, 234]
Output:
12
Input:
[1002, 2005]
Output:
10

**54.** Write a Python program to remove duplicates from a list of integers, preserving order.
Input:
[1, 3, 4, 10, 4, 1, 43]
Output:
[1, 3, 4, 10, 43]
Input:
[10, 11, 13, 23, 11, 25, 23, 76, 99]
Output:
[10, 11, 13, 23, 25, 76, 99]

**55.** Write a Python program to find the numbers that are greater than 10 and have odd first and last digits.
Input:
[1, 3, 79, 10, 4, 1, 39, 62]
Output:
[79, 39]
Input:
[11, 31, 77, 93, 48, 1, 57]
Output:
[11, 31, 77, 93, 57]

**56.** Write a Python program to find an integer exponent x such that a^x = n.
Input:
a = 2 : n = 1024
Output:
10
Input:
a = 3 : n = 81
Output:
4
Input:
a = 3 : n =
1290070078170102666248196035845070394933441741644993085810116441344597492642263849
Output:
170

**57.** Write a Python program to find the sum of the magnitudes of the elements in the array with a sign that is equal to the product of the signs of the entries.
Input:
[1, 3, -2]
Output:
-6
Input:
[1, -3, 3]
Output:
-7
Input:
[10, 32, 3]
Output:
45
Input: [-25, -12, -23]
Output: -60

**58.** Write a Python program to find the biggest even number between two numbers inclusive.
Input:
m = 12
n = 51
Output:
50
Input:
m = 1
n = 79
Output:
78
Input:
m = 47
n = 53
Output:
52
Input:
m = 100
n = 200
Output:
200

**59.** A valid filename should end in .txt, .exe, .jpg, .png, or .dll, and should have at most three digits, no additional periods. Write a Python program to create a list of True/False that determine whether candidate filename is valid or not.
Input:
['abc.txt', 'windows.dll', 'tiger.png', 'rose.jpg', 'test.py', 'win32.exe']
Output:
['Yes', 'Yes', 'Yes', 'Yes', 'No', 'Yes']
Input:
['.txt', 'windows.exe', 'tiger.jpeg', 'rose.c', 'test.java']
Output:
['No', 'Yes', 'No', 'No', 'No']

**60.** Write a Python program to find a list of all numbers that are adjacent to a prime number in the list, sorted without duplicates.
Input: [2, 17, 16, 0, 6, 4, 5]
Output: [2, 4, 16, 17]
Input: [1, 2, 19, 16, 6, 4, 10]
Output: [1, 2, 16, 19]
Input: [1, 2, 3, 5, 1, 16, 7, 11, 4]
Output: [1, 2, 3, 4, 5, 7, 11, 16]

**61.** Write a Python program to find the number which when appended to the list makes the total 0.
Input:
[1, 2, 3, 4, 5]
Output:
-15
Input:
[-1, -2, -3, -4, 5]
Output:
5
Input:
[10, 42, 17, 9, 1315182, 184, 102, 29, 15, 39, 755]
Output:
-1316384

**62.** Write a Python program to find the dictionary key whose case is different than all other keys.
Input:
{'red': '', 'GREEN': '', 'blue': 'orange'}
Output:
GREEN
Input:
{'RED': '', 'GREEN': '', 'orange': '#125GD'}
Output:
orange

**63.** Write a Python program to find the sum of the even elements that are at odd indices in a given list.
Input:
[1, 2, 3, 4, 5, 6, 7]
Output:
12
Input:
[1, 2, 8, 3, 9, 4]
Output:
6

**64.** Write a Python program to find the string consisting of all the words whose lengths are prime numbers.
Input: The quick brown fox jumps over the lazy dog.
Output: The quick brown fox jumps the
Input: Omicron Effect: Foreign Flights Won't Resume On Dec 15, Decision Later.
Output: Omicron Effect: Foreign Flights Won't On Dec 15,

**65.** Write a Python program to shift the decimal digits n places to the left, wrapping the extra digits around. If shift > the number of digits of n, reverse the string.
Input:
n = 12345 and shift = 1
Output:
Result = 23451
Input:
n = 12345 and shift = 2
Output:
Result = 34512
Input:
n = 12345 and shift = 3
Output:
Result = 45123
Input:
n = 12345 and shift = 5
Output:
Result = 12345
Input:
n = 12345 and shift = 6
Output:
Result = 54321

**66.** Write a Python program to find the indices of the closest pair from a list of numbers.
Input: [1, 7, 9, 2, 10]
Output:
[0, 3]
Input: [1.1, 4.25, 0.79, 1.0, 4.23]
Output:
[4, 1]
Input: [0.21, 11.3, 2.01, 8.0, 10.0, 3.0, 15.2]
Output:
[2, 5]

**67.** Write a Python program to find a string which, when each character is shifted (ASCII incremented) by shift, gives the result.
Input: Ascii character table
Shift = 1
Output: @rbhhbg`q`bsdqs`akd
Input: Ascii character table
Shift = -1
Output: Btdjj!dibsbdufs!ubcmf

**68.** Write a Python program to find all 5's in integers less than n that are divisible by 9 or 15.
Input:
Value of n = 50
Output:
[[15, 1], [45, 1]]
Input:
Value of n = 65
Output:
[[15, 1], [45, 1], [54, 0]]
Input:
Value of n = 75
Output:
[[15, 1], [45, 1], [54, 0]]
Input:
Value of n = 85
Output:
[[15, 1], [45, 1], [54, 0], [75, 1]]
Input:
Value of n = 150
Output:
[[15, 1], [45, 1], [54, 0], [75, 1], [105, 2], [135, 2]]

**69.** Write a Python program to create a new string by taking a string, and word by word rearranging its characters in ASCII order.
Input: Ascii character table
Output:
Aciis aaccehrrt abelt
Input: maltos won
Output:
almost now

**70.** Write a Python program to find the first negative balance from a given a list of numbers which represent bank deposits and withdrawals.
Input:
[[12, -7, 3, -89, 14, 88, -78], [-1, 2, 7]]
Output:
[-81, -1]
Input:
[[1200, 100, -900], [100, 100, -2400]]
Output:
[None, -2200]

**71.** Given a list of numbers and a number to inject, write a Python program to create a list containing that number in between each pair of adjacent numbers.
Input: [12, -7, 3, -89, 14, 88, -78, -1, 2, 7]
Separator: 6
Output:
[12, 6, -7, 6, 3, 6, -89, 6, 14, 6, 88, 6, -78, 6, -1, 6, 2, 6, 7]
Input: [1, 2, 3, 4, 5, 6]
Separator: 9
Output:
[1, 9, 2, 9, 3, 9, 4, 9, 5, 9, 6]

**72.** Write a Python program to find the indices of three numbers that sum to 0 in a given list of numbers.
Input: [12, -7, 3, -89, 14, 4, -78, -1, 2, 7]
Output:
[1, 2, 5]
Input: [1, 2, 3, 4, 5, 6, -7]
Output:
[2, 3, 6]

**73.** Write a Python program to find a substring in a given string contains a vowel between two consonants.
Input: Hello
Output:
Hel
Input: Sandwhich
Output:
San
Input: Python
Output:
hon

**74.** Write a Python program to find a string consisting of space-separated characters with given counts.
Input: {'f': 1, 'o': 2}
Output:
f o o
Input: {'a': 1, 'b': 1, 'c': 1}
Output:
a b c

**75.** Write a Python program to reorder numbers from a give array in increasing/decreasing order based on whether the first plus last element is odd/even.
Reorder numbers of a give array in increasing/decreasing order based on whether the first plus last element is odd/even.:
Input: [3, 7, 4]
Output: [3, 4, 7]
Input: [2, 7, 4]
Output: [7, 4, 2]
Input: [1, 5, 6, 7, 4, 2, 8]
Output: [1, 2, 4, 5, 6, 7, 8]
Input: [1, 5, 6, 7, 4, 2, 9]
Output: [9, 7, 6, 5, 4, 2, 1]

**76.** Write a Python program to find the index of the largest prime in the list and the sum of its digits.
Input: [3, 7, 4]
Output: [1, 7]
Input: [3, 11, 7, 17, 19, 4]
Output: [4, 10]
Input: [23, 17, 201, 14, 10473, 43225, 421, 423, 11, 10, 2022, 342157]
Output: [6, 7]

**77.** Write a Python program to convert GPAs to letter grades according to the following table:

| **GPAs** | **Grades** |
|---|---|
| 4.0: | A+ |
| 3.7: | A |
| 3.4: | A- |
| 3.0: | B+ |
| 2.7: | B |
| 2.4: | B- |
| 2.0: | C+ |
| 1.7: | C |
| 1.4: | C- |
| below: | F |

Input:
[4.0, 3.5, 3.8]
Output:
['A+', 'A-', 'A']
Input:
[5.0, 4.7, 3.4, 3.0, 2.7, 2.4, 2.0, 1.7, 1.4, 0.0]
Output:
['A+', 'A+', 'A-', 'B+', 'B', 'B-', 'C+', 'C', 'C-', 'F']

**78.** Write a Python program to find the two closest distinct numbers in a given a list of numbers.
Input: [1.3, 5.24, 0.89, 21.0, 5.27, 1.3]
Output: [5.24, 5.27]
Input: [12.02, 20.3, 15.0, 19.0, 11.0, 14.99, 17.0, 17.0, 14.4, 16.8]
Output: [14.99, 15.0]

**79.** Write a Python program to find the largest negative and smallest positive numbers (or 0 if none).
Input: [-12, -6, 300, -40, 2, 2, 3, 57, -50, -22, 12, 40, 9, 11, 18]
Output: [-6, 2]
Input: [-1, -2, -3, -4]
Output: [-1, 0]
Input: [1, 2, 3, 4]
Output: [0, 1]
Input: []
Output: [0, 0]

**80.** Write a Python program to round each float in a given list of number up to the next integer and return the running total of the integer squares.
Input: [2.6, 3.5, 6.7, 2.3, 5.6]
Output: [9, 25, 74, 83, 119]
Input: [301.1, 401.4, -23.1, 13554122.0, 10201.0101, 10000000.0]
Output: [91204, 252808, 253337, 183714223444221, 183714327525025, 283714327525025]

**81.** Write a Python program to calculate the average of the numbers a through b ( b not included ) rounded to nearest integer, in binary (or -1 if there are no such numbers).
Input: 4 , 7
Output: 0b101
Input: 11 , 19
Output: 0b1110

**82.** Write a Python program to find the sublist of numbers from a given list of numbers with only odd digits in increasing order.
Input: [1, 3, 79, 10, 4, 2, 39]
Output: [1, 3, 39, 79]
Input: [11, 31, 40, 68, 77, 93, 48, 1, 57]
Output: [1, 11, 31, 57, 77, 93]
Input: [9, -2, 3, 4, -2, 0, 2, -3, 8, -1]
Output: [-3, -1, 3, 9]

**83.** A string is happy if every three consecutive characters are distinct. Write a Python program to find two indices making a given string unhappy.

Input: Python
Output: None
Input: Unhappy
Output: [4, 5]
Input: Find
Output: None
Input: Street
Output: [3, 4]

**84.** Write a Python program to find the index of the matching parentheses for each character in a given string.

Input: ()(())
Output: [1, 0, 5, 4, 3, 2]
Input: ()()()
Output: [1, 0, 3, 2, 5, 4]
Input: ((()))
Output: [5, 4, 3, 2, 1, 0]

**85.** Write a Python program to find an increasing sequence consisting of the elements of the original list.

Input: [1, 3, 79, 10, 4, 2, 39]
Output: [1, 2, 3, 4, 10, 39, 79]
Input: [11, 31, 40, 68, 77, 93, 48, 1, 57]
Output: [1, 11, 31, 40, 48, 57, 68, 77, 93]
Input: [9, -2, 3, 4, -2, 0, 2, -3, 8, -1]
Output: [-3, -2, -1, 0, 2, 3, 4, 8, 9]

**86.** Write a Python program to find the vowels from each of the original texts (y counts as a vowel at the end of the word) from a given list of strings.

Input: ['w3resource', 'Python', 'Java', 'C++']
Output: ['eoue', 'o', 'aa', '']
Input: ['ably', 'abruptly', 'abecedary', 'apparently', 'acknowledgedly']
Output: ['ay', 'auy', 'aeeay', 'aaey', 'aoeey']

**87.** Write a Python program to find a valid substring of a given string that contains matching brackets, at least one of which is nested.

Input: ]][][[]]]
Output: [[]]
Input: ]]]]]]]]]]]]]]]]][][][][]]]]]]]]]]][[[][][[[[[[][][][]][[[[[[[[[[[[[[[[[[
Output: [[][][][]]

**88.** Write a Python program to find an integer (n >= 0) with the given number of even and odd digits.

Input: Number of even digits: 2 ,Number of odd digits: 3
Output: 22333
Input: Number of even digits: 4 ,Number of odd digits: 7
Output: 22223333333

**89.** Write a Python program to find all integers <= 1000 that are the product of exactly three primes. Each integer should represent as the list of its three prime factors.

Input: 10
Output:
[[2, 2, 2]]
Input: 50
Output:
[[2, 2, 2], [2, 2, 3], [2, 2, 5], [2, 2, 7], [2, 2, 11], [2, 3, 2], [2, 3, 3], [2, 3, 5], [2, 3, 7], [2, 5, 2], [2, 5, 3], [2, 5, 5], [2, 7, 2], [2, 7, 3], [2, 11, 2], [3, 2, 2], [3, 2, 3], [3, 2, 5], [3, 2, 7], [3, 3, 2], [3, 3, 3], [3, 3, 5], [3, 5, 2], [3, 5, 3], [3, 7, 2], [5, 2, 2], [5, 2, 3], [5, 2, 5], [5, 3, 2], [5, 3, 3], [5, 5, 2], [7, 2, 2], [7, 2, 3], [7, 3, 2], [11, 2, 2]]

**90.** For each triple of eaten, need, stock write a Python program to get a pair of total appetite and remaining.

Input:
[[2, 5, 6], [3, 9, 22]]
Output:
[[7, 1], [12, 13]]
Input:
[[2, 3, 18], [4, 9, 2], [2, 5, 7], [3, 8, 12], [4, 9, 106]]
Output:
[[5, 15], [6, 0], [7, 2], [11, 4], [13, 97]]
Input:
[[1, 2, 3], [4, 5, 6]]
Output:
[[3, 1], [9, 1]]

**91.** Write a Python program to find all n-digit integers that start or end with 2.
Input: 1
Output:
[2]
Input: 2
Output:
[12, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 32, 42, 52, 62, 72, 82, 92]
Input: 3
Output:
[102, 112, 122, 132, 142, 152, 162, 172, 182, 192, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 302, 312, 322, 332, 342, 352, 362, 372, 382, 392, 402, 412, 422, 432, 442, 452, 462, 472, 482, 492, 502, 512, 522, 532, 542, 552, 562, 572, 582, 592, 602, 612, 622, 632, 642, 652, 662, 672, 682, 692, 702, 712, 722, 732, 742, 752, 762, 772, 782, 792, 802, 812, 822, 832, 842, 852, 862, 872, 882, 892, 902, 912, 922, 932, 942, 952, 962, 972, 982, 992]

**92.** Write a Python program to start with a list of integers, keep every other element in place and otherwise sort the list.
Input: [2, 5, 6, 3, 1, 4, 34]
Output: [1, 5, 2, 3, 6, 4, 34]
Input: [8, 0, 7, 2, 9, 4, 1, 2, 8, 3]
Output: [1, 0, 7, 2, 8, 4, 8, 2, 9, 3]

**93.** Write a Python program to find the closest palindrome from a given string.
Input: cat
Output: cac
Input: madan
Output: madam
Input: radivider
Output: radividar
Input: madan
Output: madam
Input: abc
Output: aba
Input: racecbr
Output: racecar

**94.** Given a string consisting of whitespace and groups of matched parentheses, write a Python program to split it into groups of perfectly matched parentheses without any whitespace.

Input: ( ()) (((()()())) (()) ()
Output: ['(())', '(((()()()))', '(())', '()']
Input: () (( ( )() ( )) ) ( ())
Output: ['()', '(((()()()))', '(())']

**95.** Write a Python program to find a palindrome of a given length containing a given string.

Input: madam , 7
Output: madaadam
Input: madam , 6
Output: maddam
Input: madam , 5
Output: maaaam
Input: madam , 3
Output: maam
Input: madam , 2
Output: mm
Input: madam , 1
Output: aa

**96.** Write a Python program to get the single digits in numbers sorted backwards and converted to English words.

Input: [1, 3, 4, 5, 11]
Output: ['five', 'four', 'three', 'one']
Input: [27, 3, 8, 5, 1, 31]
Output: ['eight', 'five', 'three', 'one']

**97.** Write a Python program to find the following strange sort of list of numbers: the first element is the smallest, the second is the largest of the remaining, the third is the smallest of the remaining, the fourth is the smallest of the remaining, etc.

Input: [1, 3, 4, 5, 11]
Output: [1, 11, 3, 5, 4]
Input: [27, 3, 8, 5, 1, 31]
Output: [1, 31, 3, 27, 5, 8]
Input: [1, 2, 7, 3, 4, 5, 6]
Output: [1, 7, 2, 6, 3, 5, 4]

**98.** Given a string consisting of groups of matched nested parentheses separated by parentheses, write a Python program to compute the depth of each group.

Input: (()) (()) () (((()()())))
Output: [2, 2, 1, 3]
Input: () (()) () () () ()
Output: [1, 2, 1, 1, 1, 1]
Input: (((((((()))))))) () (()) (((()()())))
Output: [8, 1, 2, 3]

**99.** Write a Python program to find a string such that, when three or more spaces are compacted to a '-' and one or two spaces are replaced by underscores, leads to the target.
Input: Python-Exercises
Output:
Python Exercises
Input: Python_Exercises
Output:
Python Exercises
Input: -Hello,_world!__This_is-so-easy!-
Output:
Hello, world! This is so easy!

**100.** Write a Python program to find four positive even integers whose sum is a given integer.
Input:
n = 100
Output:
[94, 2, 2, 2]
Input:
n = 1000
Output:
[994, 2, 2, 2]
Input:
n = 10000
Output:
[9994, 2, 2, 2]
Input:
n = 1234567890
Output:
[1234567884, 2, 2, 2]