

# zekeLabs

Learn and Grow

1. Write a Python function called `compare_date` that takes as arguments two lists of two integers each. Each list contains a month and a year, in that order. The function should return -1 if the first month and year are earlier than the second month and year, 0 if they are the same, and 1 if the first month and year are later than the second. Your code should work for any legal input for month and year. Example calls and expected output are shown below:

```
>>> compare_date( [10,1995], [8,1995] )
```

```
1
```

```
>>> compare_date( [5,2010], [5,2010] )
```

```
0
```

```
>>> compare_date( [10,1993], [8,1998] )
```

```
-1
```

2. Assume `v` is a list containing numbers. Write Python code to find and print the highest two values in `v`. If the list contains only one number, print only that number. If the list is empty, print nothing. For example, if we assigned

```
v = [ 7, 3, 1, 5, 10, 6 ]
```

then the output of your code should be something like

```
7 10
```

If we are given that

```
v = [ 7 ]
```

then the output of your code should be 7

3. Consider a data, where just the name of the restaurant, the type of restaurant, and the ratings are provided. Assume these values have already been read into a list of lists of the form below:

```
restaurants = [ [ Acme, Italian, 2, 4, 3, 5],[ Flintstone, Steak, 5, 2, 4, 3, 3, 4],[ Bella Troy, Italian, 1, 4, 5] ]
```

Write a segment of Python code that prints all Italian restaurants in the restaurants list that have no ratings of value 1 and at least one rating of value 5. In the above example, Acme would be printed in the output, but Flintstone and Bella Troy would not. Flintstone is not Italian and Bella Troy has a 1 rating.

4. In the game of chess you can often estimate how well you are doing by adding the values of the pieces you have captured. The pieces are Pawns, Bishops, Knights, Rooks and Queens. Their values are

P - (P)awn, value = 1

B - (B)ishop, value = 3

K - (K)night, value = 3

R - (R)ook, value = 5

Q - (Q)ueen, value = 9

Write a Python function called `chess_score` that takes a single string as an argument and returns the combined values represented by the pieces in the string. You may assume that only P, B, K, R and Q appear in the string. You may not use any if statements and you may not use any loops. As an example,

```
print chess_score(  
BQBP  
)
```

should output the value 16 because there are 2 Bishops (3 points each), 1 Queen (9 points each), and 1 Pawn (1 point each).

5. You are given a file that contains, on each line of input, three integers separated by commas. Write a Python program that sums all of the first integers, the second integers, and the third integers, outputting the resulting sums all on one line, separated by commas. As a simple example, if the input is

2, 5, 7

3, 6, 10

1, 2, -3

2, 4, 1

Then the output should be

8, 17, 15

6. Write Python code to generate the following ranges

(a) (100; 99; 98; :: :; 0)

(b) (55; 53; 51; :: :; -1)

(c) (3; 5; 7; 9; :: :; 29)

(d) (-95; -90; -85; :: :; 85; 90)

7. Write a while loop to add all of the numbers in a list `v` until it reaches a negative number or until it reaches the end of the list. Store the sum in the variable `result`. Your code should work for any version of `v` containing only numbers. For example, the value of `result` should be 25 after the loop for both of the following lists:

```
v = [ 10, 12, 3, -5, 5, 6 ]
```

```
v = [ 0, 10, 3, 6, 5, 1 ]
```

8. Write Python code that takes a list of numbers, `v`, and outputs the positive values that are in `v` in increasing order, one value per line. If there are no positive values, then the output should be the string 'None'. You may assume there is at least one value in the list. As an example, `v = [ 17, -5, 15, -3, 12, -5, 0, 12, 22, -1 ]`

Then the output of your code should be

12

12

15

17

22

As a second example, if

```
v = [ -17, -5, -15, -3, -12, -5, 0, -12, -22, -1 ]
```

then the output should be just

None

9. What is the output of the following operations:

```
>>> mylist = [1,4,8,12,6]
```

```
>>> x = mylist.sort()
```

```
>>> print x
```

```
>>> mylist = [1,4,8,12,6]
```

```
>>> slice1 = mylist[2:4]
```

```
>>> slice1[0] = 20
```

```
>>> print slice1
```

```
>>> print mylist
```

10. Write a Python for loop to print out the values from the list v that are positive (0 is NOT a positive number).

11. What is the output of the following program?

```
def spam(a1,b1,a2,b2):
```

```
    if (a1 == a2) and (b1 > b2):
```

```
        return 1
```

```
    else:
```

```
        return 0
```

```
def egg(a1,b1,a2,b2):
```

```
    if (a1 > a2) and (b1 == b2):
```

```
        return 0
```

```
    else:
```

```
        return 1
```

```
a1 = 3
```

```
b1 = 4
a2 = 6
b2 = 4
print spam(a2, b2, a1, b1)
print egg(a1, b1, a2, b2)
c = spam(a1, b2, a2, b1)
print c
c += egg(a1, b2, a2, b1)
print c
```

12. Give the output of each of the following

(a)

```
i = 4
L = [ 0, 12, 3, 5, 2, -1 ]
while 0 <= i and i < len(L):
    if L[i] < 0:
        break
    else:
        i = L[i]
print i, L[i]
```

(b)

```
tough = 2
for i in range(2):
    s = 1
```

```
for j in range(i, tough):
```

```
s += tough
```

```
print s
```

```
print tough
```

```
tough = s
```

```
print tough
```

13. Suppose a list of words in alphabetical order has been assigned to the variable called words. For example,

we might have the assignment words = [ aardvark , abaka, expedite, experience, shoetrees, tastetest, test ]

Write code to find and output the first and the last string in words that start and end with the same letter and are at least 8 characters long. You may assume that at least one word in words satisfies this condition. You may write a function if you wish. For the above example, the output should be

6

expedite

tastetest

14. Please show the output from the following code?

```
def get_min(v):
```

```
    v.sort()
```

```
    return v[0]
```

```
def get_max(v):
```

```
    x = max(v)
```

```
    return x
```

```
v = [ 14, 19, 4, 5, 12, 8 ]
```

```
if len(v) > 10 and get_min(v) > 6:
```

```

    print "Hello"

    print v[0]

    print v[4]
else:

    print "So long"

    print v[0]

    print v[-1]
if len(v) < 10 or get_max(v):

    print get_max(v)

    print v[0]

    print get_min(v)

    print v[0]

```

15. Write code that uses a range (and NO loops) to generate the following lists:

```
v0 = [ 10, 9, 8, 7, 6, 5, 4, 3 ]
```

```
v1 = [ -10, -3, 4, 11, 18, 25, 32, 39 ]
```

16. Consider the following list of lists of strings:

```
wordy = [ [ impala, malibu, camry, jetta],[ zebra, impala, lion, impala, malibu, zebra ],[ tiger, lion,
cowboy, jet, 49er ],[ ],[ five, seven, nine] ]
```

(a) Show the output:

```
print wordy[2][1]
```

```
print wordy[1][2][3]
```

```
print len(wordy)
```

```
print len(wordy[1])
```

(b) Write a loop to print the last word of each list in wordy, stopping when either an empty list is found

or when there are no more lists. For the above example, the output should be

jetta

zebra

49er

17. Show the output of the following code. Make sure we can determine what is output and what is scratch work.

```
def remove_something(z):  
    z.remove( z[z[0]] )  
  
    v = [ 1, 8, [12, 8], hello, car ]  
  
    x = salad  
  
    if len(v[2]) >= 2:  
        if x > v[3]:  
            print One  
  
        if v[0] == 1:  
            print Three  
  
        else:  
            print Two  
  
    elif len(v) == 5:  
        print Six  
  
    else:  
        v.append(five)  
        print Ten  
  
remove_something(v)  
  
print v[1]
```



```
print v[2]

v.append(x)

print len(v)
```

18. You are given in variable x a list of lists represented as an NxN grid in which each list corresponds to one row of the grid. For example, a 4x4 grid is given by:

```
x = [[1,2,3,4],[4,3,2,1],[2,1,4,2],[2,1,4,5]]
```

Write a piece of code to print the grid in the following format with a vertical and horizontal line right in the middle:

```
1 2 | 3 4
```

```
4 3 | 2 1
```

```
----|----
```

```
2 1 | 4 2
```

```
2 1 | 4 5
```

19. Suppose you are given the scores of two athletes in various competitions, given in two separate lists. Assume there are unknown number of competitions numbered 1,2,3, etc. and the length of the two lists is the same.

```
a1 = [11,8,11,9]
```

```
a2 = [11,9,8,12]
```

For example according this to list, both athletes got a score of 11 in competition 1. Print the index of all the competitions in which a2 did better. For example, for the above lists, we would print: a2 is better in 2 4

If there is no value in which a2 is better, then you should print:

```
a2 is never better
```

20. What is the output from the following code:

```
>>> L1 = [cat, dog, hawk, tiger, parrot]
```

```
>>> print L1[1:-1]
>>> print L1[1:-2]
>>> print L1[1:-4]
>>> print L1[1:0]
>>> print L1[1:10]
>>> print L1[::-1]
>>> print L1[1:4:2]
>>> print L1[::-2]
```

---

**zekeLabs** provide Mentoring and Online & Classroom Training to IT Professionals on Cutting Edge Technologies. The Subject Matter Experts (SME) are from leading Technology Companies like Google, Yahoo!, LinkedIn and others.

To know more about us and the courses we offer visit [here](https://www.zekeLabs.com) or use link [www.zekeLabs.com](https://www.zekeLabs.com)

To know more about the SME's visit [here](https://www.zekeLabs.com/smes/) or use the link [www.zekeLabs.com/smes/](https://www.zekeLabs.com/smes/)

Subject Matter Experts From



[www.zekeLabs.com](https://www.zekeLabs.com) | +91 8095465880 | [support@zekeLabs.com](mailto:support@zekeLabs.com)