

Advanced Programming: Homework- 4

تمرین شماره چهار درس برنامه سازی پیشرفته

10.17 Exercises

1. Can a Python list hold a mixture of integers and strings?

- For instance, you can't sort a mix of integers and strings, but you can sort a list that is all integers or all strings. Otherwise you'll get a `TypeError` in the sort operation.

2. What happens if you attempt to access an element of a list using a negative index?

- In other words, you can't access an element in a list by using a negative index like `list[-1]` or `list[-2]`. However, you can use negative indexing to access elements in a string or a tuple in Python. When you use a negative index with a string or a tuple, it starts counting the positions from the end of the sequence.

3. What Python statement produces a list containing the values 45, -3, 16 and 8, in that order?

- The answer is `[45,-3,16,8]` .

4. Given the statement

```
lst = [10, -4, 11, 29]
```

(a) What expression represents the very first element of `lst`?

- `lst[0]` .

(b) What expression represents the very last element of `lst`?

- `lst[-1]` or `lst[3]` .

(c) What is `lst[0]`? ➔ `lst[0] = 10`

(d) What is `lst[3]`? ➔ `lst[3] = 29`

(e) What is `lst[1]`? ➔ `lst[1] = -4`

(f) What is `lst[-1]`? ➔ `lst[-1] = 29`

(g) What is `lst[-4]`? ➔ `lst[-4] = 10`

(h) Is the expression `lst[3.0]` legal or illegal?

- This expression is illegal because it must be a bit or an integer .

5. Given the statements

`lst = [3, 0, 1, 5, 2]`

`x = 2`

evaluate the following expressions:

(a) `lst[0]`? ➔ `lst[0] = 3`

(b) `lst[3]`? ➔ `lst[3] = 5`

(c) `lst[x]`? ➔ `lst[x] = 1`

(d) `lst[-x]`? → `lst[-x] = 5`

(e) `lst[x + 1]`? → `lst[x + 1] = 5`

(f) `lst[x] + 1`? → `lst[x] + 1 = 2`

(g) `lst[lst[x]]`? → `lst[lst[x]] = 0`

(h) `lst[lst[lst[x]]]`? → `lst[lst[lst[x]]] = 3`

6. What function returns the number of elements in a list?

- You can use the built-in `len()` method to find the length of a list. The `len()` method accepts a sequence or a collection as an argument and returns the number of elements present in the sequence or collection.

7. What expression represents the empty list?

- You can create an empty list using an empty pair of square brackets `[]`.

8. Given the list

`lst = [20, 1, -34, 40, -8, 60, 1, 3]`

evaluate the following expressions:

(a) `lst` → `lst = [20, 1, -34, 40, -8, 60, 1, 3]`

(b) `lst[0:3]` → `lst[0:3] = [20, 1, -34]`

(c) `lst[4:8]` → `lst[4:8] = [-8, 60, 1, 3]`

(d) `lst[4:33]` → `lst[4:33] = [-8, 60, 1, 3]`

(e) `lst[-5:-3]` → `lst[-5:-3] = [40, -8]`

(f) `lst[-22:3]` → `lst[-22:3] = [20, 1, -34]`

(g) `lst[4:]` → `lst[4:] = [-8, 60, 1, 3]`

(h) `lst[:]` → `lst[:] = [20, 1, -34, 40, -8, 60, 1, 3]`

(i) `lst[:4]` → `lst[:4] = [20, 1, -34, 40]`

(j) `lst[1:5]` → `lst[1:5] = [1, -34, 40, -8]`

(k) `-34 in lst` → `Ture`

(l) `-34 not in lst` → `False`

(m) `len(lst)` → `len(lst) = 8`

9. An assignment statement containing the expression `a[m:n]` on the left side and a list on the right side can modify list `a`. Complete the following table by supplying the `m` and `n` values in the slice assignment statement needed to produce the indicated list from the given original list.

Original List	Target List	Sliceindices	
		m	n
[2, 4, 6, 8, 10]	[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]		
[2, 4, 6, 8, 10]	[-10, -8, -6, -4, -2, 0, 2, 4, 6, 8, 10]		
[2, 4, 6, 8, 10]	[2, 3, 4, 5, 6, 7, 8, 10]		
[2, 4, 6, 8, 10]	[2, 4, 6, 'a', 'b', 'c', 8, 10]		
[2, 4, 6, 8, 10]	[2, 4, 6, 8, 10]		
[2, 4, 6, 8, 10]	[]		

[2, 4, 6, 8, 10]	[10, 8, 6, 4, 2]		
[2, 4, 6, 8, 10]	[2, 4, 6]		
[2, 4, 6, 8, 10]	[6, 8, 10]		
[2, 4, 6, 8, 10]	[2, 10]		
[2, 4, 6, 8, 10]	[4, 6, 8]		

✦ I could not answer the question because I did not understand it . ✦

10. Write the list represented by each of the following expressions.

(a) $[8] * 4 \rightarrow [8] * 4 = [8, 8, 8, 8]$

(b) $6 * [2, 7] \rightarrow 6 * [2, 7] = [2, 7, 2, 7, 2, 7, 2, 7, 2, 7, 2, 7]$

(c) $[1, 2, 3] + ['a', 'b', 'c', 'd']$

$\rightarrow [1, 2, 3, 'a', 'b', 'c'] = [1, 2, 3, "a", "b", "c", "d"]$

(d) $3 * [1, 2] + [4, 2]$

$\rightarrow 3 * [1, 2] + [4, 2] = [1, 2, 1, 2, 1, 2, 4, 2]$

(e) $3 * ([1, 2] + [4, 2])$

$\rightarrow 3 * ([1, 2] + [4, 2]) = [1, 2, 4, 2, 1, 2, 4, 2, 1, 2, 4, 2]$

11. Write the list represented by each of the following list comprehension expressions.

(a) `[x + 1 for x in [2, 4, 6, 8]]`

➔ `[x + 1 for x in [2, 4, 6, 8]] = [3, 5, 7, 9]`

(b) `[10*x for x in range(5, 10)]`

➔ `[10*x for x in range(5, 10)] = [50, 60, 70, 80, 90]`

(c) `[x for x in range(10, 21) if x % 3 == 0]`

➔ `[x for x in range(10, 21) if x % 3 == 0] = [12, 15, 18]`

(d) `[(x, y) for x in range(3) for y in range(4)]`

➔ `[(x, y) for x in range(3) for y in range(4)] = [(0, 0), (0, 1), (0, 2), (0, 3), (1, 0), (1, 1), (1, 2), (1, 3), (2, 0), (2, 1), (2, 2), (2, 3)]`

(e) `[(x, y) for x in range(3) for y in range(4) if (x + y) % 2 == 0]`

➔ `[(x, y) for x in range(3) for y in range(4) if (x + y) % 2 == 0] = [(0, 0), (0, 2), (1, 1), (1, 3), (2, 0), (2, 2)]`

12. Provide a list comprehension expression for each of the following lists.

(a) `[1, 4, 9, 16, 25]` ➔ `[y*y for y in range(1,6)]`

(b) `[0.25, 0.5, 0.75, 1.0, 1.25, 1.5]` ➔ `[z/4 for z in range (1,7)]`

(c) `[('a', 0), ('a', 1), ('a', 2), ('b', 0), ('b', 1), ('b', 2)]`

➔ `[(x,y) for x in 'ab' for y in range(0,3)]`

13.If lst is a list, what expression indicates whether or not x is a member of lst?

- `x in lst` , means x is in the lst .
- `x not in lst` , means x is not in the lst .

14. What does reversed do?

- The `reversed()` method computes the reverse of a given sequence object and returns it in the form of a list.



پاسخ سوال های ۱۵ تا ۲۳ را در Colab نوشته ام و در GitHub قرارداده ام
و در اینجا فقط روی سوال ها را نوشته ام .

15. Complete the following function that adds up all the positive values in a list of integers. For example, if list a contains the elements 3 , -3 , 5 , 2 , -1, and 2, the call `sum_positive(a)` would evaluate to 12, since $3+5+2+2 = 12$. The function returns zero if the list is empty.

```
def sum_positive(a):  
  
    # Add your code...
```

16. Complete the following function that counts the even numbers in a list of integers. For example, if list a contains the elements 3 , 5, 4, -1 , and 0, the call `count_evens(a)` would evaluate to 2, since a contains two even numbers: 4 and 0. The function returns zero if the list is empty. The function does not affect the contents of the list.

```
def count_evens(lst):
```

```
# Add your code...
```

17. Write a function named `print_big_enough` that accepts two parameters, a list of numbers and a number. The function should print, in order, all the elements in the list that are at least as large as the second parameter.

18. Write a function named `next_number` that accepts a list of integer values. All the elements in the list are unique, and all elements in the list are greater than or equal to one. (The caller must ensure that these conditions are met before passing the list to `next_number`.) The `next_number` function should return the smallest positive integer not in the list. (Note that 1 is the smallest positive integer.)

As examples,

- `next_number([5, 3, 1])` would return 2
- `next_number([5, 4, 1, 2])` would return 3

- `next_number([2, 3])` would return 1
- `next_number([])` would return 1

19. Write a function named `reverse` that reorders the contents of a list so they are reversed from their original order. `a` is a list. Note that your function must physically rearrange the elements within the list, not just print the elements in reverse order.

20. Write a Python program that creates the matrix

```
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
```

and assigns it to the variable `m`. Pretty print `m` to ensure the contents are correct. Next, reassign `m[2][4]` to 0, and print `m` again to ensure your code modified the correct element.

21. Provide five different ways to create the list `[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]` and assign it to the variable `lst`.

22. In a square 2D list the number of rows equals the number of columns. Write a function that accepts a square 2D list and

returns True if the left to right contents of any row equals the top to bottom contents of any column. If no row matches any column, the function returns False.

23. We can represent a Tic-Tac-Toe board as a 3 X 3 grid in which each position can hold one of the following three strings: "X", "O", or " ". Write a function named `check_winner` that accepts a 3 X 3 list as a parameter. If "X" appears in a winning Tic-Tac-Toe pattern, the function should return the string "X". If "O" appears in a winning Tic-Tac-Toe pattern, the function should return the string "O". If no winning pattern exists, the function should return the string " ".