

Welcome to 420-SF2-RE!

Section 1	Mon: 10:00 – 11:30	Wed: 10:00 – 11:30	Thurs: 8:00 – 10:00
Section 2	Mon: 14:30 – 16:00	Tue: 11:30 – 13:30	Thurs: 14:30 – 16:00

Data Structures and Object Oriented Programming

Week 5

Louisa Harutyunyan

Office: 3F.33

Contact: via MIO

Suggested Office Hours: Monday 11:30 – 12:30, Wednesday 12:30 – 13:30

ConcepTest

What is the output after the following snippet of code is executed?

```
num = [1, 2, 3, 4, 5]
for i in range(len(num)):
    num[i] = num[i] * 2
num = num + (6, 7)
print(num)
```

- A) [1, 2, 3, 4, 5, 6, 7]
- B) [2, 4, 6, 8, 10, 6, 7]
- C) [2, 4, 6, 8, 10], (6, 7)
- D) [2, 4, 6, 8, 10], [6, 7]
- E) This code produces a TypeError

ConcepTest

What is the output after the following snippet of code is executed?

```
num = [1, 2, 3, 4, 5]
num += (2, 3)
for i in range(len(num)):
    num[i] = num[i] * 2
print(num)
```

- A) [1, 2, 3, 4, 5, 2, 3]
- B) [2, 4, 6, 8, 10, 4, 6]
- C) [2, 4, 6, 8, 10], (4, 6)
- D) [2, 4, 6, 8, 10], [4, 6]
- E) This code produces a TypeError

ConcepTest

What prints?

```
t = ('Feb', 17, 2025)
t2 = ('Mar', 7, 2025)
print(t + t2)
t[0] = 'Apr'
print(t)
```

ConcepTest

What prints?

```
t = ('apple', 17.3, [99, 98])  
t2 = (33.77, [7, 8])  
print(t[2] + t2[1])
```

ConcepTest

What prints?

```
t = ('apple', 17.3, [99, 98])  
tup = (t[2], ) + (3, ['apricot', 'pear'])  
tup[2] = ['tomato']  
print(tup)
```

ConceptTest

What prints?

```
t = ('apple', 17.3, [99, 98])  
num = t[2]  
num[1] = 0  
print(t)
```

ConcepTest

What prints?

```
t = ('apple', 17.3, [99, 98])  
lst = t[2]  
t[2][0] = 12  
lst.append('pear')  
print(t)  
print(lst)
```


ConceptTest

What is the output of the following code?

```
lst = ['cat', 'dog', 'p']  
lst[1] = lst[0][0]  
print(len(lst) + len(lst[1]))
```

- A) 3
- B) 4
- C) 7
- D) 8
- E) No output; there is an error

ConcepTest

What are the values of lst1 and lst2 respectively?

```
lst1 = [1, 2, 3]
lst2 = [lst1] + [lst1]
lst2[0] = 99
```

- A) lst1: [1, 2, 3] lst2: [99, 2, 3, 1, 2, 3]
- B) lst1: [99, 2, 3] lst2: [99, 2, 3, 1, 2, 3]
- C) lst1: [1, 2, 3] lst2: [99, [1, 2, 3]]
- D) lst1: [99, 2, 3] lst2: [99, [1, 2, 3]]
- E) None of the above

ConcepTest

What is the value of `lst` after this code runs?

```
lst = [2, 3, 5, 7, 11, 17, 19]
lst.pop(3)
lst.remove(2)
```

- A) [2, 7, 11, 17, 19]
- B) [5, 7, 11, 17, 19]
- C) [2, 3, 11, 17, 19]
- D) [3, 5, 11, 17, 19]
- E) Nothing; the code produces an error

ConcepTest

What is the output of the following code?

```
lst = [20, 15.5, 12.0, 7, 33]
minimum = min(lst)
maximum = max('whale')
print(minimum, maximum)
```

- A) 7 e
- B) 7 w
- C) 7.0 w
- D) No output; there is an error

ConcepTest

Consider the code below that runs on an arbitrary list `lst`. What does it do?

```
lst = [...] # a list
good = True
i = 0
while i < len(lst)-1 and good:
    good = lst[i] < lst[i+1]
    i = i + 2
print(good)
```

- A) prints whether the list is sorted
- B) prints whether each pair of elements in the list is in increasing order
- C) prints whether the first pair of elements in the list is in increasing order
- D) works fine, but crashes on the empty list
- E) None of the above

ConceptTest

What is the output of the following code?

```
lst = [[2, 3, 4], [1, 1]]
x = 0
for i in range(len(lst)):
    for j in range(len(lst[0])):
        x = x + lst[i][j]
print(x)
```

- A) 2
- B) 7
- C) 11
- D) This code produces an error

ConcepTest

What is the list referred to by `a` after the following snippet of code is executed?

```
a = [[1, 2, 3], [4, 5]]  
b = a[:]  
b.append(8)
```

- A) `[[1, 2, 3], [4, 5]]`
- B) `[[1, 2, 3], [4, 5], 8]`
- C) `[[1, 2, 3], [4, 5, 8]]`
- D) `[[1, 2, 3], [4, 5], [8]]`

ConcepTest

What are the values of `lst1` and `lst2` respectively?

```
lst1 = [1, 2, 3]
lst2 = [lst1] + [lst1]
lst2[0][1] = 99
```

- A) `lst1`: [1, 2, 3] `lst2`: [[1, 99, 3], [1, 2, 3]]
- B) `lst1`: [1, 2, 3] `lst2`: [[99, 2, 3], [1, 2, 3]]
- C) `lst1`: [99, 2, 3] `lst2`: [[99, 2, 3], [1, 2, 3]]
- D) `lst1`: [1, 99, 3] `lst2`: [[1, 99, 3], [1, 99, 3]]
- E) None of the above

ConceptTest

In what situation does the following code fail?

```
# read list and value from user
# print index of first occurrence of value in list
# or -1 if value is not in list
```

```
lst = input().split()
```

```
value = input()
```

```
i = 0
```

```
num = lst[i]
```

```
while num != value:
```

```
    i = i + 1
```

```
    num = lst[i]
```

```
if i < len(lst):
```

```
    print(i)
```

```
else:
```

```
    print(-1)
```

- A) It never fails
- B) It fails when the list is empty
- C) It fails when the value is not found in the list
- D) It fails in both conditions B and C