Sequences and Containers

Aditya Iyer

How is your week going so far?

1. What is something that is going well for you?
2. What is something that is not going so well for you?

Outline of Today's Section

Total time: 50 mins

Mini Lecture: 10 mins (we'll be covering lists, but probably won't have time for dictionaries)

Worksheet: 40 mins

- Q1: Lists WWPD
- Q2: Lists Environment Diagrams
- Q3 : List Comprehensions
- Q4: Snapshot

[Optional] : Exam Level Question

Some useful Methods

1. Len(): This method is used to get the number of elements of a list

```
A = [1,2,3,4,5]
B = len(A)
```

2. Range(a,b): Gives you the sequence of numbers from a to b-1. This is useful in iterations .

```
for i in range (4,8):
    print(i)
```

Repetition and Combination

Repetition:

```
I = [1,2,3,4,5]
I*2 --> [2,4,6,8,10]
```

Combination;

```
[-1,0]+I--> [-1,0,1,2,3,4,5]
```

List Comprehension

Can you predict the output?

```
[i for i in range(2,7)]
```

Try this one now!

```
[[i] for i in range (0,10) if i%2 == 0]
```

Important point to note!

What is the difference between the two?

```
lst += [element]
```

```
lst = lst + [element]
```

Box and Pointer Diagram

Each element of a list, can either have a primitive value or it has a pointer to another reference value.

Examples:

```
l1=[1,2,3,[5,6]]
```

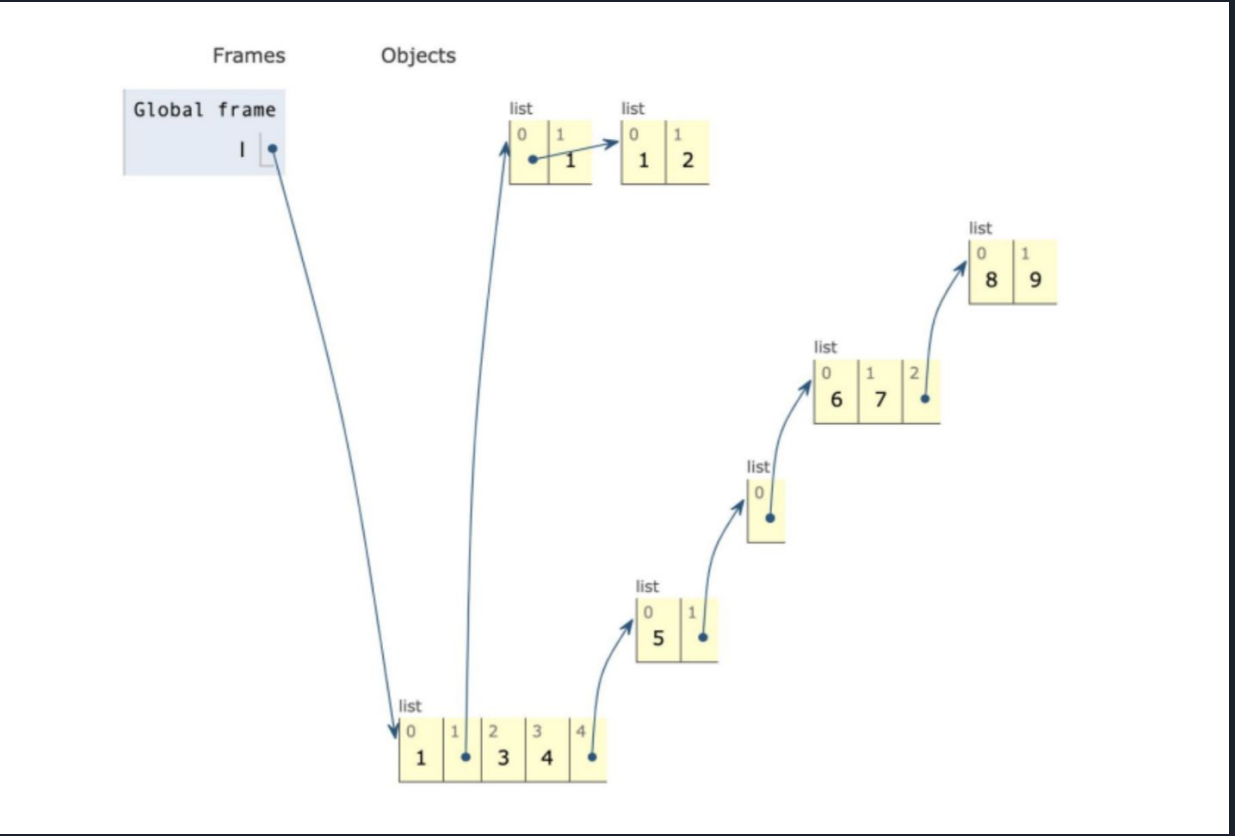
```
l= [1,[[1,2],1],3,4[5,[[6,7,[8,9]]]]]
```

Take a minute or two to try and draw the box and pointer diagrams

Note: This is a very complex example! It's okay if you can't solve it, you likely won't see this level of

complexity on the exam. The point is to drive home the concept.

Box and Pointer Diagram



Lists Slicing! (Credits: CSM Content Team)

Very important!! List slicing creates a new list.

- Slicing: creates a copy of all or part of a sequence
- Syntax: seq[<start index>: <exclusive end index>: <step size>]
    - Default step size is 1
    - Default start index is 0
    - Default end index is len(seq)
    - A negative step goes backward from end of sequence
- seq[:] copies an entire sequence
- seq[1:] copies all but first element
- seq[::-1] copies seq in reverse order
- seq[:5:2] copies every other item starting at the beginning and ending at item 4

Some more List Methods!

list. append(x): adds an element x to the end of the list

list. extend(sequence) : adds a sequence (another list for example) to the end of the list

list. insert (index,x) : inserts item x at specified index

list. remove(x): removes the first index of x (no return value)

list.pop(index): removes and returns the element at specified index. (if no index is specified, then it removes the last element)

Click here to try out these methods in Python Tutor to get a better feel for this!

Previous Exam Question(Fall 2019, Q2)

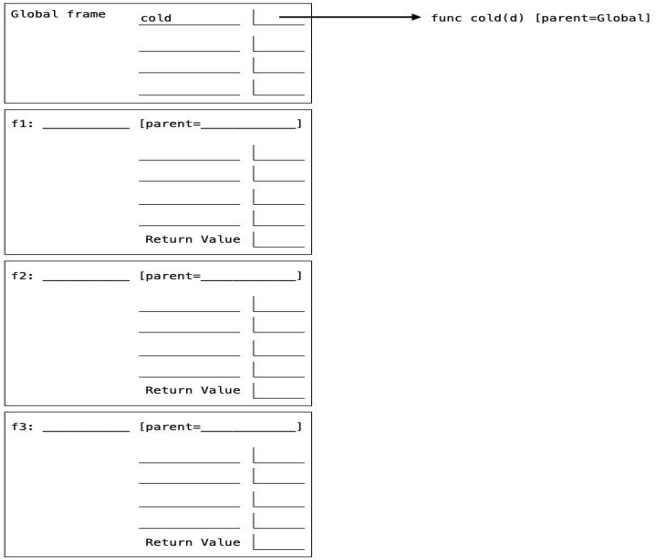
2. (8 points) Protect the Environment

Fill in the environment diagram that results from executing the code on the right until the entire program is finished, an error occurs, or all frames are filled.

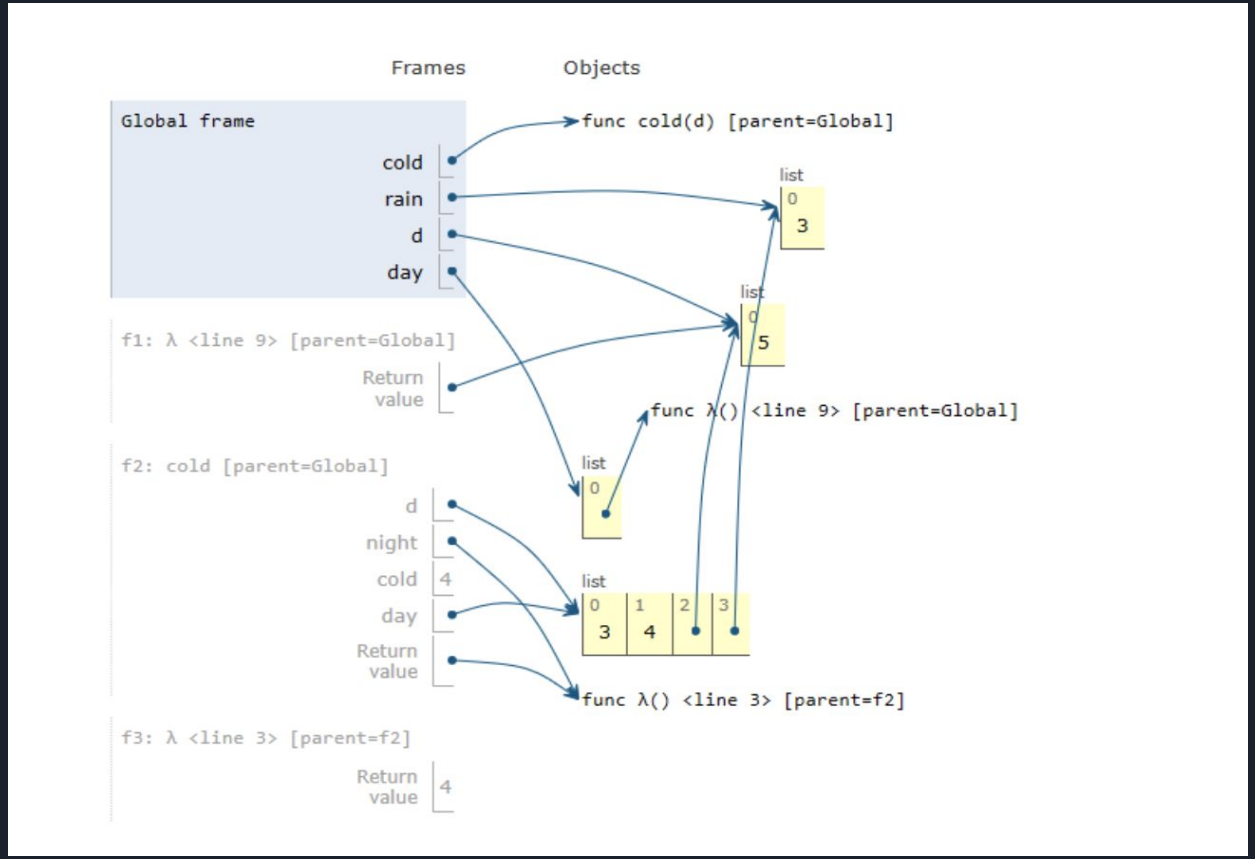
Important: You may not need to use all of the spaces or frames. Do not draw frames for calls to built-in functions, such as `len`. A complete answer will:

- Use box-and-pointer notation for lists.
- Add all missing names and parent annotations.
- Add all missing values created or referenced.
- Show the return value for each local frame.

```
1  def cold(d):
2      day = rain[:1]
3      night = lambda: len(day)
4      cold = rain.pop()
5      day = d
6      return night
7
8  rain = [3, 4]
9  d, day = [5], [lambda: d]
10 cold(rain + [day[0](), rain])()
```



Previous Exam Solution



Anonymous Feedback Form