Advanced Python Old Exam Questions

Any code you write should be short and "pythonic" (Not C written in Python!)

1. (a) Suppose we execute the following statements

```
D = { }
S = 'one'
L = ['one', 'two']
T = ('one', 'two')
```

For each of the following, indicate whether it is a valid Python statement; if it is not valid, explain why.

```
a) D[S] = 'bacon'          valid

b) D[T] = 'bacon'          valid

c) D[L] = 'bacon'          invalid; lists are not immutable (thus unhashable)

d) D[5] = 'bacon'          valid
```

1(b) Suppose we have the following definitions.

```
def func1(L):
     L[0] = 5

def func2(L):
     L = L[:-1] + [8]
```

For each of the print statements below, show the output.

```
K = [3,6,5,2]
func1(K)
print K
```

```
[5, 6, 5, 2]  # The local variable is set to reference the argument list, the
              # argument list can be modified
```

```
J = [8,4,9,1,6]
func2(J)
print J
```

```
[8,4,9,1,6]  # The local variable is set to reference the argument list,
             # but then is changed to refer to a new list
```

Advanced Python Old Exam Questions

2. Write a Python program that inputs a sentence and then prints the words of the sentence in reverse order. Words are separated by whitespace.

Example (user input underlined):

```
Enter a sentence: you can cage a swallow can't you
you can't swallow a cage can you
```

```
s = input(Enter a sentence: )
L = s.split()
L.reverse()
print ‘ ‘.join(L)
```

or

```
s = input(Enter a sentence: )
print ‘ ‘.join(reversed(s.split()))
```

3. Write a Python function half_backwards(L) with parameter a list L that returns the list consisting of the second half of L followed by the first half. If the length of the list is odd, the middle element is considered as part of the second half of the list.

Hint: a//b is the integer quotient obtained by dividing a by b.

Examples

```
L = [1,2,3,4]
K = half_backwards(L)
# Now K == [3,4,1,2]
```

```
L = [1,2,3,4,5]
K = half_backwards(L)
# now K == [3,4,5,1,2]
```

```
def half_backwards(L):
  return L[:len(L)//2] + L[len(L):]
```

Advanced Python Old Exam Questions

4. Write a python program that creates a dictionary whose keys are names (strings) and whose value for a key is a list of the favorite foods of the person with that name. Your program should prompt for and input a name and a favorite food until the user enters an empty string for the name.

After creating the dictionary, the program prints one line per dictionary entry consisting of the name, the word "likes", and the favorite foods of the person with that name as show in the example below (user input underlined):

```
Enter a sequence of names followed by a favorite food, terminating input by
hitting enter for the name.

Name: John
Food: chocolate

Name: Karen
Food: dates

Name: John
Food: merlot

Name: Karen
Food: sauternes

Name: John
Food: steak

Name:

John likes chocolate, merlot, and steak
Karen likes dates and sauternes
```

```
D = {}
name = input("Name: ")
while(name):
    food = input("Food: ")
    D[name] = food
    name = input("Name: ")

for name in D:
    print( ', '.join(D[name][-1]))
```

5. a) For an object to be used as a dictionary key it must be **hashable**

(b) Could t = (2,[3,4],4) be used as a dictionary key? **No. Explain your answer: while it is immutable, it contains an element that is not immutable; therefore it is not hashable**

6. Write a list comprehension that constructs a list of the squares of odd integers less than a given integer n.

```
OddsSquared = [ y*y for y in range(1000) if y % 2 == 0 ]
```

Advanced Python Old Exam Questions

7. You are to write a function that constructs a list of the integers contained in a text file with name *fname*. Ideally, the file should exist and contain digit strings separated by white space. But this may not be the case and your code segment should take this into account. If errors occur, your function should print the corresponding error message and return None. Otherwise, it returns the integer list. And no matter what occurs, any opened file should be closed. Your code should use a `try-except-finally` block and may require more than one Exception possibility. Hint: if you are going to use f for the file object, set it to `None` before attempting to open the file.

```
def getIntList(fname):
    fh = None
    try:
        fh = open(fname)
        return [int(k) for k in fh.read().split()]
    except FileNotFoundError as e1:
        print(e1)
    except TypeError as e2:
        print(e2)
    finally:
        if fh is not None:
            fh.close()
```

8(a) The immutable version of `list` is `tuple`. What type is the immutable version of `set`?

frozenset

(b) Using the `set` type, write a single line statement that will produce a list L of the unique values of an arbitrary list K.

Example: if `K = [ 8, 2, 3, 8, 2, 6, 3]`, then `L` could be `[2, 8, 3, 6]`.

`L` **`= list(set(K))`**

(c) Assuming the above, fill in the sorting key to rearrange the values in L so that they in the order of first appearance in K.

`L.sort(key =` **`K.index`**`)`

Advanced Python Old Exam Questions

9. Define lists as follows: **L_1 = [ ], L_2 = [1, 0], L_3 = [2, 3]**

For each expression below, fill in the value returned when the expression is evaluated.

L_1 and L_2	**L_1**
L_1 or L_2	**L_2**
L_2 and L_3	**L_3**
L_3 and L_2	**L_2**
~L_1	**True**

10. Write a function named avg that accepts an arbitrary number of arguments (assumed to be numbers) and returns their average value

```
def avg(*k):
    return sum(k)/len(k)
```

11. Suppose L is a list of 2-tuples of float objects. Write a **single line** that will sort the list in order of the sum of the entries of each tuple.

Example:

Before sorting: `L = [(1.0,3.2), (-1.0,1.0), (1.0,1.0)]`
After sorting: `L = [(-1.0,1.0), (1.0,1.0), (1.0,3.2)]`

```
L.sort(key = sum)
```

12. Write a list comprehension that produces the same list as the following code:

```
L = []
for k in K:
     if k % 2 == 0:
            L.append(k//2)
```

```
L = [ k//2 for k in K if k %2 == 0]
```

]

13. Write the code for the function `get_int(msg)` which is intended to input an integer from the user and return that object. Input from the user is obtained by calling `input(msg)`, where `msg` is the prompt. If an error occurs, print the error message and return; otherwise return the integer. Recall that `input()` returns a string.

```
def get_int(msg):
   try:
      return int(input(msg))
   except ValueError as e:
      print(e)
```

14. Suppose L = [R, S, T, U] is a list of sets. Using ordinary set notation, using union (∪) and intersection (∩), describe the following set:

```
{ x for x in L[0] if all([x in A for A in L[1:]])}
```

R ∩ S ∩ T ∩ U

15. Let D be a dictionary whose values are all hashable. Construct a dictionary E that maps distinct values of D to the list of all keys of D that map to the given value.

```
E = {}
for v in D.values():
    E[v] = [k for k in D if D[k] == v]
```

or

```
E = {v: [k for k in D if D[k] == v] for v in D.values()}
```