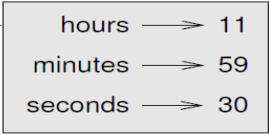
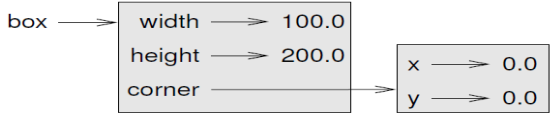


Internal Assessment Test 3 – May 2019

Sub:	Python Application Programming				Sub Code:	15CS664	Branch :	CSE		
Date:	15/05/2019	Duration:	90 min's	Max Marks:	50	Sem/Sec :	6 th A/B / C	OBE		
<u>Answer any FIVE FULL Questions</u>								MAR KS	CO	RB T
1 (a)	<p>What is a pure function? Write a python function, duration (time1, time2) that returns duration of an event for the following TIME class. An instance of the class is given below:</p> <div style="text-align: center;">  </div> <ul style="list-style-type: none"> • Pure function doesn't produce side effects like, • Prompting input from user(E.g.: input statements), Modifying the argument passed, Generating some outputs (E.g.: print statements) • In turn it communicates with the calling function only through return values (i.e. return statement) • E.g.: len <p>Assuming a 24 hour scale of time, class time:</p> <pre>hr,min,sec=0,0,0 def __init__(self,a,b,c): self.hr,self.min,self.sec=a,b,c def sub_time(t1,t2): s=time() s.hr=t2.hr-t1.hr s.min=t2.min-t1.min s.sec=t2.sec-t1.sec return s t1=time(11,45,56) #movie start time t2=time(13,30,34) #movie end time duration=sub_time(t1,t2) print(duration)</pre>							6	CO4	L3
(b)	<p>Explain the need of the self-argument in the methods of any class with an example.</p> <ul style="list-style-type: none"> • Self-variable refers to the object itself and it refers to the attributes of an instance • Self is very similar to 'this' keyword in java • It is by default the first argument of any method of a class and this is how class method are differentiated from other functions • While calling the method value for self-need not be passed. It just acts as a reference to the instance • Name it as self always as it is the name highly recommended 							4	CO4	L2

	<ul style="list-style-type: none"> • But still other names are also accepted <p>Example :</p> <pre>class abc: def sumof(self,a,b): print(self.a+self.b) obj1=abc() obj1.sumof(3,5) # internally call will be made as abc.sumof(obj1,3,5) where self denotes the object</pre> <ul style="list-style-type: none"> • E.g.: lets assume a class 'abc' having a method sum of which will read two numbers and returns the sum • The definition would be 'defsumof(self, a,b):' • When the method is called it should be called like obj1.sumof(3,5) by just passing values for a and b where obj1 is the object for the class 'abc'. • The python interpreter reads it like abc.sumof(obj1,3,5) • And this is why self argument need not be passed with any value 			
2 (a)	<p>Differentiate deep copy from shallow copy with an example.</p> <ul style="list-style-type: none"> • A shallow copy constructs a new compound object and then inserts references to the child objects found in original • i.e shallow copying means copying reference of an object to another object • Hence the changes made in either one of the object gets reflected in other object also copy.copy(object) will create a shallow copy of passed object <p>Eg: import copy li=[[1,2,3],[4,5,6]] ob1=copy.copy(li) li[0][1]=22 print(li) print(ob1)</p> <ul style="list-style-type: none"> • A deep copy constructs a compound new object and then recursively populate the object with copy of its child objects found in original • deep copying means copy of an object gets copied to another object • Hence the changes made in either one of the object doesn't get reflected in other object as they both are individual objects with different references • copy.deepcopy(object) will create a shallow copy of passed object <p>e.g.: import copy li=[[1,2,3],[4,5,6]] ob1=copy.deepcopy(li) li[0][1]=22 print(li,ob1) print(ob1)</p>	4	CO4	L2
(b)	<p>Explain __init__ and __str__ methods with an example python program</p> <p>Init Method:</p> <ul style="list-style-type: none"> • Method gets automatically invoked when an object is created for a class. Even 	6	CO4	L2

	<p>the arguments of init methods can have default values/ keyword arguments.</p> <pre>class time: def __init__(self, hour=0,min=0,sec=0): self.hour=hour self.min=min self.sec=sec</pre> <p>str Method:</p> <ul style="list-style-type: none"> • <code>__str__</code> is a special method like <code>__init__</code> that is supposed to return a string representation of an object. • An explicit call is not required for this method • The function gets invoked automatically when we try to print an object <pre>class time: def __init__(self, hour=0,min=0,sec=0): self.hour=hour self.min=min self.sec=sec def __str__(self): return ("%0.2d:%0.2d:%0.2d" %(self.hour, self.min, self.sec))</pre> <pre>t1=time(3,45,52) print(t1)</pre>			
3 (a)	<p>What is embedded SQL? Explain the importance of SQLite data base. Write a python code to establish a database connection to EmpDb and display the total gross salary paid to the employees working in quality control department. Assume that employee table has already been created and exists in EmpDb. The fields of employee are: (EmpID, Department, GrossSalary)</p> <ul style="list-style-type: none"> • Embedded SQL is a method of combining the computing power of a programming language and the database manipulation capabilities of SQL. Embedded SQL statements are SQL statements written inline with the program source code of the host language. • SQLite is already built into python and very well suited for data manipulation problems <pre>import sqlite3 conn = sqlite3.connect('EmpDb') cur = conn.cursor() print("Tracks:") cur.execute('SELECT GrossSalary FROM Emp where Department='Qualitycontrol') for row in cur: sum=sum+row print('Total Gross salary:',sum) conn.commit() cur.close()</pre>	10	CO5	L2

<p>4 (a)</p>	<p>Define a class called Point with attributes x and y. Instantiate an object called corner. Define a class called RECTANGLE with attributes: width, height and corner and instantiate 'box' object. Write member method to print rectangle and findCenter() of a rectangle.</p>  <pre> class point: x=0 y=0 class Rectangle: def __init__(self, h, w): self.height = h self.width = w def printrectangle(rect): print(rect.height, rect.width) def find_center(rect): p=point() p.x=rect.corner.x+rect.width/2 p.y=rect.corner.y+rect.height/2 return p rect = Rectangle(12, 10) rect.corner=point() rect.corner.x=10.0 rect.corner.y=20.0 center=rect.find_center() print(center) print(rect.printrectangle()) </pre>	6	CO4	L3
<p>(b)</p>	<p>Define Polymorphism. Explain with an example python program.</p> <ul style="list-style-type: none"> Polymorphism means taking many forms Example : The program discussed to find the frequency of letters in a string (histogram) i.e. using dictionary as a counter will work for all types like string, list, tuple etc., Real polymorphism is when you discover that the function we defined works for other types that we have never planned for. <pre> def histogram(s): d=dict() for i in s: if i not in d: d[i]=1 else: d[i]+=1 return d </pre>	4	CO4	L2
<p>5 (a)</p>	<p>Write a note on Google GeoEncoding Web Services. Using Python supported libraries, demonstrate with a snippet code.</p> <ul style="list-style-type: none"> Google has an excellent web service that allows us to make use of their large 	10	CO5	L2

database of geographic information.

- We can submit a geographical search string like “Ann Arbor, MI” to their geocoding API and have Google return its best guess as to where on a map we might find our search string and tell us about the landmarks nearby

```
import urllib.request, urllib.parse, urllib.error
import json
import ssl

api_key = False
# If you have a Google Places API key, enter it here
# api_key = 'AIzaSy__IDByT70'
# https://developers.google.com/maps/documentation/geocoding/intro

if api_key is False:
    api_key = 42
    serviceurl = 'http://py4e-data.dr-chuck.net/json?'
else :
    serviceurl = 'https://maps.googleapis.com/maps/api/geocode/json?'

# Ignore SSL certificate errors
ctx = ssl.create_default_context()
ctx.check_hostname = False
ctx.verify_mode = ssl.CERT_NONE

while True:
    address = input('Enter location: ')
    if len(address) < 1: break

    parms = dict()
    parms['address'] = address
    if api_key is not False: parms['key'] = api_key
    url = serviceurl + urllib.parse.urlencode(parms)

    print('Retrieving', url)
    uh = urllib.request.urlopen(url, context=ctx)
    data = uh.read().decode()
    print('Retrieved', len(data), 'characters')

    try:
        js = json.loads(data)
    except:
        js = None

    if not js or 'status' not in js or js['status'] != 'OK':
        print('==== Failure To Retrieve ====')
        print(data)
        continue

    print(json.dumps(js, indent=4))

    lat = js['results'][0]['geometry']['location']['lat']
```

	<pre> lng = js['results'][0]['geometry']['location']['lng'] print('lat', lat, 'lng', lng) location = js['results'][0]['formatted_address'] print(location) </pre>			
6 (a)	<p>Write short notes on XML. Design python program to retrieve a node present in XML tree.</p> <ul style="list-style-type: none"> • XML resembles HTML and was developed for describing the data while HTML was to display the data • XML tags are not predefined • XML is a structured language where we have a flexibility to create our own tags • It takes tree structure to describe the data <p>XML Node:</p> <pre> <person> <name>Chuck</name> <phone type="intl"> +1 734 303 4456 </phone> <email hide="yes" /> </person> </pre> <p>E.g.:</p> <pre> import xml.etree.ElementTree as ET data = "<person><name>Chuck</name><phone type='intl'>+1 734 303 4456 </phone><email hide='yes' /></person>" tree = ET.fromstring(data) print('Name:', tree.find('name').text) print('Attr:', tree.find('email').get('hide')) </pre>	6	CO5	L2
(b)	<p>Explain urllib package to retrieve contents of web page with example.</p> <ul style="list-style-type: none"> • Its is the URL handling module for python • urllib package has several modules like • urllib.request: opening and reading • urllib.parse: for parsing URLs (i.e. reading html code and generating DOM tree structure) • urllib.error: to handle exceptions • urllib.robotparser: parsing robot.txt files <p>e.g.:</p> <pre> import urllib.request fhand= urllib.request.urlopen('http://data.pr4e.org/romeo.txt') for line in fhand: print(line.decode().strip()) </pre>	5	CO5	L2
7(a)	<p>What is JSON? Explain with code snippet, parsing JSON in Python.</p> <ul style="list-style-type: none"> • JSON is a lightweight format for storing and transporting data. • JSON is often used when data is sent from a server to a web page. • Python's syntax for dictionaries and lists influenced the syntax of JSON 	5	CO5	L2

- So the format of JSON is nearly identical to a combination of Python lists and dictionaries
- Its like a key value pair.
- All the tags and attributes of tags are a colon separated key value pair

```
import json

data = """
[
  { "id" : "001",
    "x" : "2",
    "name" : "Chuck"
  },
  { "id" : "009",
    "x" : "7",
    "name" : "Brent"
  }
]"""

info = json.loads(data)
print('User count:', len(info))

for item in info:
    print('Name', item['name'])
    print('Id', item['id'])
    print('Attribute', item['x'])
```

(b) Write program to retrieve image over HTTP using methods from the socket class.

```
import socket
import time

HOST = 'data.pr4e.org'
PORT = 80
mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
mysock.connect((HOST, PORT))
mysock.sendall(b'GET http://data.pr4e.org/cover3.jpg HTTP/1.0\r\n\r\n')
count = 0
picture = b""

while True:
    data = mysock.recv(5120)
    if len(data) < 1: break
    #time.sleep(0.25)
    count = count + len(data)
    print(len(data), count)
    picture = picture + data

mysock.close()

# Look for the end of the header (2 CRLF)
pos = picture.find(b"\r\n\r\n")
```

5

CO5

L2

	<pre>print('Header length', pos) print(picture[:pos].decode()) # Skip past the header and save the picture data picture = picture[pos+4:] fhand = open("stuff.jpg", "wb") fhand.write(picture) fhand.close()</pre>			
8(a)	<p>Define web scraping and build a program to extract data from home page of cmrit site</p> <ul style="list-style-type: none"> • Web scraping means designing a program that pretends like browser such that it extracts large amounts of data from the web pages • Why are we extracting data from web pages? • Say some webpages may not allow us to download data from their site. In such cases we can scrape the webpage using python which is the most wanted skill for any data analyst or data scientist <pre>import urllib.request, urllib.parse, urllib.error img = urllib.request.urlopen('http://cmrit.ac.in') fhand = open('a.txt', 'wb') size = 0 while True: info = img.read(100000) if len(info) < 1: break size = size + len(info) fhand.write(info) print(size, 'characters copied.') fhand.close()</pre>	5	CO5	L2
(b)	<p>Write a program to extract all links from a web page using RE or BeautifulSoup.</p> <pre># Search for link values within URL input import urllib.request, urllib.parse, urllib.error import re import ssl # Ignore SSL certificate errors ctx = ssl.create_default_context() ctx.check_hostname = False ctx.verify_mode = ssl.CERT_NONE url = input('Enter - ') html = urllib.request.urlopen(url).read() links = re.findall(b'href="(http[s]?://.*?)"', html) for link in links: print(link.decode()) or # To run this, you can install BeautifulSoup</pre>	5	CO5	L2


```
# https://pypi.python.org/pypi/beautifulsoup4
```

```
# Or download the file
```

```
# http://www.py4e.com/code3/bs4.zip
```

```
# and unzip it in the same directory as this file
```

```
import urllib.request, urllib.parse, urllib.error
```

```
from bs4 import BeautifulSoup
```

```
import ssl
```

```
# Ignore SSL certificate errors
```

```
ctx = ssl.create_default_context()
```

```
ctx.check_hostname = False
```

```
ctx.verify_mode = ssl.CERT_NONE
```

```
url = input('Enter - ')

```

```
html = urllib.request.urlopen(url, context=ctx).read()
```

```
soup = BeautifulSoup(html, 'html.parser')
```

```
# Retrieve all of the anchor tags
```

```
tags = soup('a')
```

```
for tag in tags:
```

```
    print(tag.get('href', None))
```