KENDRIYA VIDYALAYA SANGATHAN MUMBAI REGION
PREBOARD EXAMINATION (2023-24)
CLASS XII
SUBJECT: COMPUTER SCIENCE (083)

Time allowed: 3 Hours Maximum Marks: 70

SET-1

MARKING SCHEME

Ques No	Question	Marks
	SECTION A	
1	True	1
2	b. 11 digits	1
3	b. **	1
4	(A)hAPPY#HOUR1122#33	1
5	b.7,18	1
6	a. PAN	1
7	b. Error	1
8	(d) ("f","o","obar")	1
9	False True	1
10	a) 10#20# or b) 10#20#30#40#50# or c) 10#20#30#	1
11	a. Wireless transmission or d. unguided transmission	1
12	c. global b	1
13	True	1
14	b. distinct	1
15	message	1
16	c. tell()	1
17	(A)	1
18	(A)	1
	SECTION B	
19	(i) eXtensible Markup Language, Simple Mail Transfer Protocol (ii) HTML(Hyper text mark Up language) • We use pre-defined tags • Static web development language – only focuses on how data looks • It use for only displaying data, cannot transport data • Not case sensistive XML (Extensible Markup Language) • we can define our own tags and use them • Dynamic web development language – as it is used for transporting and storing data • Case sensitive OR (i) Baud is the number bits carrying in one second. (ii) https (Hyper Text Transfer Protocol Secure) is the protocol that uses SSL (Secure Socket Layer) to encrypt data being transmitted over the Internet. Therefore, https helps in secure browsing while http does not.	1+1=2
20	The following Python code is supposed to print the largest word in a sentence but there are few errors. Rewrite the code after removing syntax and logical errors and underline each corrections made. Str=input("Enter a sentence")	2

	word=Str.split()
	print(word)
	maxlen=0
	largest=""
	for i in word:
	  l=len(i)
	  if(l>maxlen):
	  largest=i
	  maxlen=l
	print(largest)	

21	def COUNTWORDS( ):
	  fin=open('PYTHON.TXT', 'r')
	  count=0
	  for line in fin:
	  for i in line.split( ):
	  if i[0].isupper( ):
	  count+=1
	  print(count)
	  fin.close( )
	
	OR
	
	def ALCount( ):
	  f = open('STORY.TXT' , 'r')
	  countA = 0
	  countL = 0
	  for line in f:
	  if line[0] == 'A' or line[0] == 'a' :
	  countA+=1
	  if line[0] == 'L' or line[0] == 'l' :
	  countL+=1
	  print('A or a : ', countA)
	  print('L or l : ', countL)
	  f.close()
	
	(½ mark for correctly opening and closing the file
	½ for correct loop
	½ for correct if statement
	½ for correctly incrementing count
	)
	
	Note: Any other relevant and correct code may be marked
		2

22	[11, 10, 9, 8, 7, 4]	2
23	Write the Python statement for each of the following tasks using BUILT-IN functions/methods only: (i) L1.insert(3,400) (ii) message.endswith('.') OR import statistics print(statistics.mode(employeesalary))	2
24	Alter table customer add custid integer primary key;	

	Insert into customer values('Nandini','Management',45600,555); OR Alter table dance drop category; Alter table dance add typecca varchar(10) not null;	
25	300 @ 200 300 # 100 150 @ 100 300 @ 150	
	SECTION C	
26	us@2023 4	3
27	(i) Sum(Loan_Amount) 1200000 (ii) Max(Interest) 4500 (iii) Count(*) 2	1*3=3

28	def countH():
	f=open('Para.txt','r')
	rec=f.readlines()
	count=0
	for line in rec:
	if line[0]=='H':
	count+=1
	print('The line count =',=count)
	f.close()
	(½ mark for correctly opening and closing the file
	½ for correct loop
	1 for correct condition
	½ for correct if statement
	½ for correctly incrementing count
	)
	
	Note: Any other relevant and correct code may be marked
	
	OR
	def countmy():
	f=open('DATA.TXT','r')
	rec=f.read()
	words=rec.split()
	for w in words:
	if w=='my':
	count+=1
	print('my occurs',count,'times')
	f.close()
	(½ mark for correctly opening and closing the file
	½ for correct loop
	1 for correct condition
	½ for correct if statement
	½ for correctly incrementing count
		3

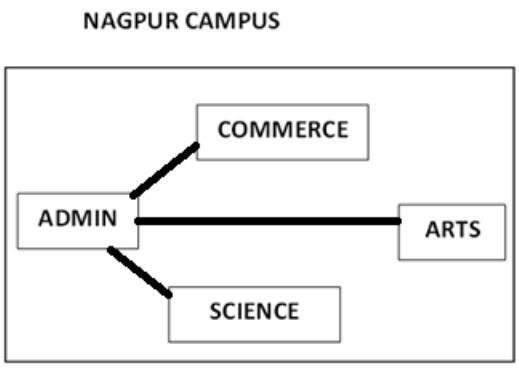
	) Note: Any other relevant and correct code may be marked	
29	Consider the table ACTIVITY given below: ACODE | ACTIVITYNAME | PARTICIPANTSNUM | PRIZEMONEY | SCHEDULEDATE 1001 | Relay Name | 16 | 10000 | 23-Jan-2004 1002 | High Jump | 10 | 12000 | 12-Dec-2003 1003 | Shot Put | 12 | 8000 | 14-Feb-2004 1005 | Long Jump | 12 | 9000 | 01-Jan-2004 1008 | Discuss Throw | 10 | 15000 | 19-Mar-2004 Based on the given table, write SQL queries for the following: (i) Select * from activity where prizemoney >=9000; (ii) Update activity set prizemoney=prizemoney*1.05 where scheduledate >'01-Mar-2004'; (iii) Delete from activity where participantsnum<12;	3

30	Rajiv has created a dictionary containing employee names and their salaries as key value pairs of 6 employees.
	Write a program, with separate user defined functions to perform the following operations:
	● Push the keys (employee name) of the dictionary into a stack, where the corresponding value (salary) is less than 85000.
	● Pop and display the content of the stack.
	For example:
	If the sample content of the dictionary is as follows:
	
	Emp={"Ajay":76000, "Jyothi":150000, "David":89000, "Remya":65000, "Karthika":90000, "Vijay":82000}
	
	The output from the program should be:
	Vijay Remya Ajay
	Emp={"Ajay":76000, "Jyothi":150000, "David":89000, "Remya":65000, "Karthika":90000, "Vijay":82000}
	Stack=[]
	def Push():
	for ename in Emp:
	if Emp[ename]<85000:
	Stack.append(ename)
	def Pop():
	if Stack==[]:
	print("Stack Empty")
	else:
	top = len(Stack) – 1
	for a in range (top -1, -1,-1):
	print(Stack[a],end=" ")
		3

	SECTION D	
31	i. Select distinct Qty from garment; ii. Select sum(Qty) from garment group by CCode having count(*)>1;	1*4=4

	iii. Select GNAME, CNAME, RATE from garment g, cloth c where g.ccode=c.ccode and Qty>100; iv. Select avg(Rate) from garment where rate between 1000 and 2000;	

32	def Insert():
	L=[]
	while True:
	ClockID = input("Enter Clock ID = ")
	ClockName = input("Enter Clock Name = ")
	YearofManf = int(input("Enter Year of Manufacture = "))
	price = float(input("Enter Price = "))
	R = [ClockID, ClockName, YearofManf, price]
	L.append(R)
	ans = input("Do you want to enter more records (Y/N)=")
	if ans.upper()=='N':
	break
	import csv
	fout = open('watch.csv','a',newline='')
	W = csv.writer(fout)
	W.writerows(L)
	fout.close()
	print("Records successfully saved")
	def Delete():
	ClockID = input("Enter Clock ID to be removed = ")
	found = False
	import csv
	fin = open('watch.csv','r')
	R = csv.reader(fin)
	L = list(R)
	fin.close()
	for i in L:
	if i[0] == ClockID:
	found=True
	print("Record to be removed is:")
	print(i)
	L.remove(i)
	break
	if found==False:
	print("Record not found")
	else:
	fout = open('watch.csv','w',newline='')
	W = csv.writer(fout)
	W.writerows(L)
	fout.close()
	print("Record Successfully Removed")
	
	Insert() function
	½ mark for correct data input and making list
	½ mark for correctly opening file
	1 mark for correctly writing record
	Delete() function
	½ mark for correctly copying data in list
		4

	½ mark for correctly identifying record and removing it from the list ½ mark for correctly showing not found message ½ mark for correctly re-writing remaining records	
	SECTION E	
33	**a. The most suitable building to house the server is ADMIN building because it has maximum number of computers and as per 80:20 rule this building will have the maximum amount of network traffic.** **½ mark for correct answer** **½ mark for correct justification**  **b.** **1 mark for correct diagram** **c. iii. Video Conferencing** **1 mark for correct diagram** **d.** **i. Switch/Hub will be placed in every building to provide network connectivity to all devices inside the building.** **ii. Repeater will not be required as there is not cable running for more than 100 meters.** **½ mark each for each correct reason** **e. The device/software that can be installed for data security and to protect unauthorized access is Firewall.** **1 mark for correct answer**	1*5=5

34	(i) r+ mode: • Primary function is reading • File pointer is at beginning of file • if the file does not exist, it results in an error w+ mode: • primary function is writing • if the file does not exist, it creates a new file. • If the file exists, previous data is overwritten • File pointer is at the beginning of file. (minimum two differences should be given)
	(ii)
	def copyData():
	fobj=open(“STUDENT.DAT”,”rb”)
	fobj1=open(“HIGHACHIEVERS.DAT”,”wb”)
	cnt=0
	try:
	while True:
	data=pickle.load(fob)
	print(data)
	if data[2]>60:
	pickle.dump(data,fobj1)
	cnt+=1
	except:
	fobj.close()
		2+3=5

	fobj1.close()
	return cnt
	(½ mark for correctly opening and closing files ½ mark for correct try and except block ½ mark for correct loop 1 mark for correctly copying data 2+3=5
	OR
	(i) Text files: • Extension is .txt • Data is stored in ASCII format that is human readable • Has EOL character that terminates each line of data stored in the text files Binary Files • Extension is .dat • Data is stored in binary form (0s and 1s), that is not human readable.
	(ii)
	def findType(atype):
	fobj=open("BANK.DAT", "rb")
	try:
	while True:
	data=pickle.load(fobj)
	if data[2]==atype:
	print("Account Number", data[0])
	print("Customer Name", data[1])
	print("Account Type", data[2])
	except:
	fobj.close()
	½ mark for correct return statement ½ mark for correctly opening and closing files ½ mark for correct try and except block ½ mark for correct loop ½ mark for correct if statement 1 mark for correctly displaying data	

35	(i) Foreign Key is a non key attribute for which values are derived from the primary key of another table. (½ marks for definition and ½ for any suitable example) (ii) import mysql.connector mycon=mysql.connector.connect(host='localhost',user='root', passwd='KVS@123',databse='KV') mycur=mycon.cursor() fn=input("Enter flight number") s=input("Enter source') d=input("Enter Destination") f=int(input("Enter fare of flight")) query="insert into flight values('{}','{}', '{}',{}).format(fn,s,d,f) mycur.execute(query) mycon.commit() print("Data added successfully") mycon.close() ½ mark for importing correct module 1 mark for correct connect() ½ mark for correctly accepting the input 1 ½ mark for correctly executing the query ½ mark for correctly using commit()) Note: Any other correct logic may be marked OR (i) Primary key is one or more attribute that can uniquely identify the tuple in relation. Alternate key is a candidate key which is not selected as a primary key. Only one primary key exist in the a relation. Alternate keys may be more than one. (½ Marks for each point of difference). (ii) import mysql.connector mycon=mysql.connector.connect(host='localhost',user='root', passwd='KVS@123',databse='Sports') mycur=mycon.cursor()	1+4=5

	query=”select * from game where No_of_Participants>{}”.format(10) mycur.execute(query) data=mycur.fetchall() for rec in data: print(rec) mycon.close() (½ mark for importing correct module 1 mark for correct connect() 1 mark for correctly executing the query ½ mark for correctly using fetchall() 1 mark for correctly [15] displaying data)	