ANNEXURE –C

DAV PUBLIC SCHOOLS, ODISHA ZONE

NAME OF THE EXAM: Half Yearly Examination, SUBJECT: Computer Science
CLASS: XII

MARKING SCHEME
SET - 1

QS NO	VALUE POINT	MARKS ALLOTTED	PAGE NO. OF NCERT TEXT BOOK/ EXAMPLAR
1	False	1 mark for correct answer	Sumita Arora Pg-3
2	(a) str1[::-6]	1 mark for correct answer	Sumita Arora Pg-45
3	(a) dict_exam.update(dict_result)	1 mark for correct answer	Sumita Arora Pg-70
4	(c) 15.0	1 mark for correct answer	Move fast with Sumita Arora Pg-14
5	(d) [20, 37, 61, 90, 98]	1 mark for correct answer	Sumita Arora Pg-57
6	(c) 'TWO'	1 mark for correct answer	CBSE SQP 2021-22
7	c) def calcu (a=1,b=2,c=3):	1 mark for correct answer	Sumita Arora Page-105
8	(a) (14, 0)	1 mark for correct answer	Sumita Arora Page-111
9	(a) dt = f.readlines();print(dt[3])	1 mark for correct answer	Sumita Arora Pg-188
10	(c) both iii and iv	1 mark for correct answer	CBSE SQP 2021-22
11	(c) Every line ends with a new line character	1 mark for correct answer	CBSE SQP 2021-22
12	(b)fobj.readline()	1 mark for correct answer	Sumita Arora Pg-191
13	(b) Fp.seek (offset, 0)	1 mark for correct answer	Sumita Arora Pg-213
14	(c) with open('Employee.dat','ab') as ob: pickle.dump(E,ob)	1 mark for correct answer	Sumita Arora Pg-209
15	(d) IndexError: pop from empty list	1 mark for each correct answer	Sumita Arora Pg-56
16	Grace of one mark to be awarded	1 mark for correct answer	Sumita Arora Pg-354
17	(d) A is False but R is True.	1 mark for correct answer	Sumita Arora Page-113

18	(a) Both A and R are true and R is the correct explanation for A.	1 mark for correct answer	Sumita Arora Page-218
19	i) AAA ii) a **OR** i) Moonsionsion ii) i	1 mark for each correct answer	Sumita Arora Page-17 (Practical Book)
20	**def** Tot(Number)**:** Sum = 0 for C in range(1, Number + 1) : Sum += C **return** Sum print(**Tot(3)**) **OR** def Check(): n = int(input("Enter a number ")) for k in range (1,n//2)**:** : missing if k*k == n: = missing print("Square root = ",k) break if k == n // 2-1: / missing print("Not a perfect square ") " missing Check()	½ mark for each correction ½ mark for each correction	Sumita Arora Page-108
21	lst = eval(input("enter the list = ")) print ("New list =", [lst[-1]] + lst[0 : -1]) (Any other alternative method may be considered) **OR** D1={'A':[1,2,3],'B':[4,5,6]} D2={} for i in D1: D2[i]=sum(D1[i]) print(D2)	½ mark 1½ marks ½ mark ½ mark 1 mark	Sumita Arora Page-53 Sumita Arora Page-494 (Class-XI)
22	wORLDCUPp#2#2	1 mark for wORLDCUP 1 mark for p#2#2	Sumita Arora Page-80
23	(i) ng is i 12 (ii) F.seek(10,1)	½ mark ½ mark 1 mark	Sumita Arora Page-212
24	Output: quit talking and begin doing	2 marks for correct line of	Sumita Arora Page-233

	OR Output: talking and	output.	

25	stack=[]
	def PUSH(Lst):
	for i in Lst:
	if i%2 == 0 and i%10 == 6:
	stack.append(i)
	print(stack)
	L=[11,33,66,92,16,45,56,26]
	PUSH(L)
		½ mark for correct loop
		1 mark for checking condition
		½ mark for appending element to stack
			Sumita Arora Page-356

26	2 {'Apple': 2, 'Banana': 3}	1 mark for 2 2 marks for {'Apple': 2, 'Banana': 3}	APC Books Page-32
27	(i) 8 (ii) ('Ch', 'a', 'ndrayan-3') (iii) {'A': 'L1', 'd': 'L1', 'i': 'L1', 't': 'L1', 'y': 'L1', 'a': 'L1'}	1 mark 1 mark 1 mark	Sumita Arora Page-48

28	def SHOW(s):
	D={'UPPER CASE':0,'LOWER CASE':0,'DIGIT':0,'SPECIAL CHARACTER':0}
	for i in s:
	if i.isupper():
	D['UPPER CASE']+=1
	elif i.islower():
	D['LOWER CASE']+=1
	elif i.isdigit():
	D['DIGIT']+=1
	else:
	D['SPECIAL CHARACTER']+=1
	return D
	OR
	def CHECK(L):
	ctr=0
	print("Names of the cities which start with A =")
	for i in L:
	if i[0]=='A':
	print(i)
	if i[0]!='A':
	ctr+=1
	print("Total number of cities not starting with A =",ctr)
		½ for fn header
		½ for defining dictionary
		2 marks for calculation
		(OR)
		½ for fn header
		½ for loop
		1 mark for printing cities starts with 'A'
		1 mark for counting cities those don't start with 'A'
			Together With Page-46
			Together With Page-55

29	(i) English (ii) PE Biology English Physics Computer Sc	1 mark 1 mark	Sumita Arora Page-65

	Chemistry (iii) PE	1 mark	

30	R={"UK":"EUROPE", "INDIA":"ASIA",
	"CHINA":"ASIA",
	"EGYPT":"AFRICA","CUBA":"AMERICA",
	"JAPAN":"ASIA"}
	stack=[]
	def PUSH():
	for i in R:
	if R[i]=="ASIA":
	stack.append(i)
	def POP():
	if stack==[]:
	print("Underflow")
	else:
	while stack!=[]:
	print(stack.pop(),end=" ")
	PUSH()
	POP()
		1.5 marks for correct PUSH() and 1.5 marks for correct POP()	Sample Question Paper 2021-22

31	i) ValueError ii) ZeroDivisionError iii) Finally iv) Denominator should not be zero JOB OVER... GO GET SOME REST	1 mark 1 mark 1 mark 1 mark	Sumita Arora Supplement

32	def Words():
	f=open("STORY.txt",'r')
	data=f.readlines()
	line=1
	totw=0
	for i in data:
	c=0
	word=i.split()
	for j in word:
	c=c+1
	sum+=len(j)
	print('No.of words in line',line,'=',c)
	totw+=c
	line+=1
	avgw=sum/totw
	print('Total words in file = ',totw)
	print('Average word length =',avgw
	f.close()
	
	Note: Any other relevant and correct code may be marked
	
	No marks to be deducted for not calculating average word length of the file.
		½ mark for correctly opening and closing file
		½ mark for readlines()
		½ mark for correct loop to read line wise
		½ mark for correct logic to count the total word line wise
		½ mark for correctly incrementing line counts
		½ mark for calculating total words
		1 mark for correctly displaying the outputs
			Sumita Arora Page-187

33	i) fruitful function ii) var1,var2=Local scope p,q,r,s =Global scope iii) var1,var2=Formal Parameter iv) 250@300 v) 50@245.5	1 mark ½ mark ½ mark 1 mark ½ mark ½ mark 1 mark	Sumita Arora Page-107

34	(i) Serialization is the process of converting input object (i.e. list, tuples, dictionaries etc) into machine readable byte stream that gets stored in the binary file.
	In **wb** mode, it overwrites the file if the file exists. In case, the file does not exists, it will create a new file for writing. In **ab** mode the file pointer is placed at the end of the file to add new records. In case the file does not exist, it will create a new file for writing.
	(ii)
	def SEARCH():
	f=open("EMP.dat","rb")
	try:
	while True :
	s=pickle.load(f)
	found=0
	for i in s:
	if i[2]>=50000 and i[2]<=80000:
	print(i)
	found=1
	if found==0:
	print("No such record exists ...")
	except EOFError:
	f.close()
	
	(1/2 mark for correctly opening the file)
	(1/2 mark for correctly reading from binary file)
	(1 mark for correctly writing the loop)
	
	OR
	def Prize_Winner():
	f=open("PRIZE.dat","rb")
	try:
	while True:
	R=pickle.load(f)
	if R[2]==1:
	print(R)
	except:
	f.close()
		1 mark
		
		1 mark
		
		½ mark
		
		½ mark
		½ mark
		½ mark
		½ mark
		
		½ mark
		½ mark
		½ mark
		½ mark
		½ mark
			Sumita Arora
			Page-210
			
			Sumita Arora
			Page-210
			
			APC Books
			Page-210

35	(i) "BOOKS.CSV", "a" (ii) csv.writer(csvf) (iii)cw.writerow(['Title','Author','Price']) (iv)csv.reader(csvf)	1 mark 1 mark 1 mark 1 mark 1 mark	Sumita Arora Page-218

	(v) a. r[0][0]=='R' **OR** (i) Line 1 : csv (ii) Line 2 : a (iii) Line 3 : reader (iv) Line 4 : close() (v) Line 5 : Aman 123@456 Anis aru@nima Raju myname@FRD	1 mark 1 mark 1 mark 1 mark 1 mark 1 mark	Sumita Arora Page-218