

## DAV PUBLIC SCHOOLS, ODISHA ZONE

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

**MARKING SCHEME  
SET - 2**

QS NO	VALUE POINT	MARKS ALLOTTED	PAGE NO. OF NCERT TEXT BOOK/ EXAMPLAR
1	(a) str1[::-6]	1 mark for correct answer	Sumita Arora Pg-45
2	True	1 mark for correct answer	Sumita Arora Pg-6
3	(a) dict_exam.update(dict_result)	1 mark for correct answer	Sumita Arora Pg-70
4	(c) True	1 mark for correct answer	Move fast with Sumita Arora Pg-14
5	(c) 'TWO'	1 mark for correct answer	CBSE SQP 2021-22
6	(d) [20, 37, 61, 90, 98]	1 mark for correct answer	Sumita Arora Pg-57
7	c) None	1 mark for correct answer	Sumita Arora Page-114
8	(d) 512	1 mark for correct answer	Sumita Arora Page-110
9	(a) dt = f.readlines();print(dt[3])	1 mark for correct answer	Sumita Arora Pg-188
10	(b) r+	1 mark for correct answer	Sumita Arora Pg-186
11	(c) Every line ends with a new line character	1 mark for correct answer	CBSE SQP 2021-22
12	(d) fobj.readlines()	1 mark for correct answer	Sumita Arora Pg-188
13	(d) f.seek(50, 0)	1 mark for correct answer	Sumita Arora Pg-213
14	(d) IndexError: pop from empty list	1 mark for correct answer	Sumita Arora Pg-56
15	(c) with open('Employee.dat', 'ab') as ob: pickle.dump(E,ob)	1 mark for each correct answer	Sumita Arora Pg-209
16	(d) ST.remove(-1)	1 mark for correct answer	Sumita Arora Pg-56,57
17	(d) A is False but R is True.	1 mark for correct answer	Sumita Arora Page-218

18	(a) Both A and R are true and R is the correct explanation for A.	1 mark for correct answer	Sumita Arora Page-113
19	i) AND ii) n  <b>OR</b> i) Moonsionsion ii) i	1 mark for each correct answer	Sumita Arora Page-44
20	(i) ng is i 12 (ii) F.seek(10,1)	½ mark ½ mark 1 mark	Sumita Arora Page-212
21	lst = eval(input("enter the list = ")) print ("New list =", [lst[ -1 ] ] + lst[0 : -1 ] )  (Any other alternative method may be considered)  <b>OR</b> D1={'A':[1,2,3],'B':[4,5,6]} D2={} for i in D1: D2[i]=sum(D1[i]) print(D2)	½ mark 1½ mark  ½ mark ½ mark 1 mark	Sumita Arora Page-53  Sumita Arora Page-494 (Class-XI)
22	<b>fUNnpYTHON\$\$. OR fUNnpYTHON.\$</b>	1 mark for fUNnpY 1 mark for THON.\$	Sumita Arora Page-47
23	<b>def Tot(Number):</b> Sum = 0 for C in range(1, Number + 1) : Sum += C <b>return</b> Sum print( <b>Tot(3)</b> ) <b>OR</b> def Check(): n = int(input("Enter a number ")) for k in range (1,n//2) : if k*k == n: print("Square root = ",k) break if k == n // 2-1: print("Not a perfect square ")  Check()	½ mark for each correction  ½ mark for each correction	Sumita Arora Page-108
24	stack=[] def PUSH(Lst): for i in Lst: if i%2 == 0 and i%10 == 6: stack.append(i) print(stack)	½ mark for correct loop 1 mark for checking condition ½ mark for	Sumita Arora Page-356

	L=[11,33,66,92,16,45,56,26] PUSH(L)	appending element to stack	
25	Output: quit talking and begin doing OR Output: talking and	2 marks for correct line of output.	Sumita Arora Page-233
26	a) Reg.keys() b) Reg.values() c) Reg.pop('Mukesh')	1 mark 1 mark 1 mark	Sumita Arora Pg-66,67
27	(i) 3 (ii) print(T[0][0][1]) (iii) TP = 100, or TP = (100,)	1 mark 1 mark 1 mark	Sumita Arora Page-58,59
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  <b>OR</b> 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  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	CS = {"Raj" : 80 , "Anu" : 91 , "Vishwa" : 95 , "Moni" : 80 , "Govind" : 90} stack = [] def push(CS): for i in CS : if CS[i] >= 90: stack.append(i)  def pop(): while True : if stack == []: print("UNDERFLOW") break else : print(stack.pop())	1.5 marks for correct PUSH() and 1.5 marks for correct POP()	Sample Question Paper 2021-22

30	(i) English (ii) <table border="1" data-bbox="320 192 541 562" style="margin-left: 40px;"> <tr><td>PE</td></tr> <tr><td>Biology</td></tr> <tr><td>English</td></tr> <tr><td>Physics</td></tr> <tr><td>Computer Sc</td></tr> <tr><td>Chemistry</td></tr> </table> (iii) PE	PE	Biology	English	Physics	Computer Sc	Chemistry	1 mark 1 mark  1 mark	Sumita Arora Page-65
PE									
Biology									
English									
Physics									
Computer Sc									
Chemistry									
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	<pre>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()</pre> <p>Note: Any other relevant and correct code may be marked</p> <p style="color: red;">No marks to be deducted for not calculating average word length of the file.</p>	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	1 mark ½ mark ½ mark 1 mark ½ mark	Sumita Arora Page-107						

	v) 50@245.5	½ mark 1 mark	
34	<p>(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 <b>wb</b> mode, it overwrites the file if the file exists. In case, the file does not exist, it will create a new file for writing. In <b>ab</b> 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.</p> <p>(ii) def SEARCH():     f=open("EMP.dat","rb")     try:         while True :             s=pickle.load(f)             found=0             for i in s:                 if i[2]&gt;=50000 and i[2]&lt;=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)</p> <p style="text-align: center;"><b>OR</b></p> <p>def Prize_Winner():     f=open("PRIZE.dat","rb")     try:         while True:             R=pickle.load(f)             if R[2]==1:                 print(R)     except:         f.close()</p>	<p>1 mark</p> <p>1 mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p> <p>½ mark</p>	<p>Sumita Arora Page-210</p> <p>Sumita Arora Page-210</p> <p>APC Books Page-210</p>
35	<p>(i) "BOOKS.CSV", "a" (ii) csv.writer(csvf) (iii) cw.writerow(['Title','Author','Price']) (iv) csv.reader(csvf) (v) a. r[0][0]=='R'</p> <p style="text-align: center;"><b>OR</b></p> <p>(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</p>	<p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p> <p>1 mark</p>	<p>Sumita Arora Page-218</p> <p>Sumita Arora Page-218</p>

--	--	--	--