

INDIAN SCHOOL MUSCAT
HALF YEARLY EXAMINATION
SEPTEMBER 2019

SET C

CLASS XII

Marking Scheme – COMPUTER SCIENCE [THEORY]

Q.NO.	Answers	Marks (with split up)
1(a)	iostream.h iomanip.h (½ Mark each for writing correct header file)	1
(b)	<pre> typedef int Count; //Error 1, Error 2 void main() { Count C; int K; //OR Count K; //Error 3 cout<<"Enter the count:"; cin>>C; for (K = 1; K<=C; K++) //OR for (int K = 1; K<=C; K++) //Error 3 //OR for (Count K = 1; K<=C; K++) //Error 3 cout<< C << "*" << K <<endl; //Error 4 //OR cout<<C * K<<endl; //Error 4 } </pre> (½ Mark for correcting each Error and rewriting the statement correctly)	2
(c)	Price*Qty float Address+One do (½ Mark for each correct name)	2
(d)	The output expected from the program is (iii) 20#17# (1 mark , working should be shown) Minimum Value of Alter = 10 (½ Mark) Maximum Value of Alter = 11 (½ Mark)	2
(e)	400*40 200*10 8000*400 (½ Mark for writing each correct value)	3
(f)	<u>Output</u> 10*15*93*20*73*12* (3 Marks for correct Output) (½ Mark for writing each correct value)	3
2(a)	Data Encapsulation: Wrapping up of data and functions together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and functions together	2

	<p>in a single unit. (½ Mark)</p> <p>Data Hiding: Keeping the data in private visibility mode of the class to prevent it from accidental change is known as Data Hiding. . (½ Mark)</p> <p>Any example. (1 Mark)</p>	
(b)	<p>An object is an identifiable entity with some characteristics and behavior. It represents an entity that can store data and its associated functions. (1 Mark)</p> <p>A class is a group of objects that share common properties and relationships. It represents a group of similar objects. (1 Mark)</p>	2
3(a)	<p>correct comparison between default arguments and function overloading (1 Mark each)</p>	2
(b)	<p>(i) callout(56); //function 4 (ii) callout('p',77.2F); //function 2 (iii) callout(83,77.33f); //function 3 (iv) callout(4,66,'x'); //function 1 (½ Mark each)</p>	2
4(a)	<p>protected Visibility Mode (1 Mark)</p> <p>The members in protected visibility modes are not accessible to objects but are accessible in derived classes.</p> <p>public visibility mode: (1 Mark)</p> <p>Members of a class declared under this visibility are accessible inside the class (in member functions of the class) as well as by the Objects of that class (in any non member function of the program, prototyped / defined after the class declaration).</p>	2
(b)	<pre> class ENVIRONMENT { char City[20]; int PMLevel; char Health[15]; void AssignHealth(); public: void In(); void Out(); }; void ENVIRONMENT::AssignHealth() { if (PMLevel<=50) strcpy(Health,"Healthy"); else if (PMLevel<=100) strcpy (Health,"Moderate"); else strcpy(Health,"Unhealthy"); } void ENVIRONMENT::In() { gets(City); cin>>PMLevel; AssignHealth(); } void ENVIRONMENT::Out() { cout<<City<<PMLevel<<Health<<endl; } </pre> <p>(½ Mark for declaring class header correctly)</p>	4

	(½ Mark for declaring data members correctly) (1 Mark for defining AssignHealth () correctly) (½ Mark for taking inputs of City and PMLevel in In()) (½ Mark for invoking AssignHealth() inside In()) (½ Mark for defining Out() correctly) (½ Mark for correctly closing class declaration with a semicolon ;)		
5(a)	(i) CAMERA B(“SONY”, 20000.25) ; // any valid values (ii) CAMERA(TV &temp) { strcpy (company, temp.company) ; price = temp.price ; } (1 Mark each for the correct answer)		2
(a)	OR Correct Difference - (1 Mark each)		
(b)	Constructor	Destructor	2
	Name of the constructor function is same as that of class	Name of the destructor function is same as that of class preceded by ~	
	Constructor functions are called automatically at the time of creation of the object	Destructor functions are called automatically when the scope of the object gets over	
	Constructor can be overloaded	Destructor cannot be overloaded	
	Constructor is used to initialize the data members of the class	Destructor is used to de- initialize the data members of the class	
	Correct Differences(Any 2)- (1 Mark each)		
6(a)	Hierarchical inheritance- when many subclasses inherit from a single base class. Eg: Base class C ↙ ↘ Derived class A Derived class B In a multilevel inheritance a class is derived from an immediate base class. Eg: class A (Base class of B) ↑ class B (Derived class of A, Base class of C) ↑ class C (Derived lower most class) (1 Mark each for the correct answer)		2
(b)	(i)None of data members are accessible from objects belonging to class AUTHOR. (1 Mark)		4

	<p>(ii) Enter(), Show() (1 Mark)</p> <p>(iii) Data members: Voucher_No, Sales_Date, Salary (½ Mark) Member function: Sales_Entry(), Sales_Detail(), Enter(), Show(), Register(), Status() (½ Mark)</p> <p>(iv) 66 (1 Mark)</p>	
7(a)	<pre>void DispMorN () { ifstream File ("NOTES.TXT"); char STR[80]; while(File.getline(STR,80)) { if(STR[0]=='M' STR[0]=='N') cout<<STR<<endl; } File.close(); }</pre> <p>(½ Mark for opening NOTES. TXT correctly) (1 Mark for reading each Line (Whichever method adopted) from the file) (1 Mark for checking lines starting with 'M' or 'N') (½ Mark for displaying the lines)</p>	3
(b)	<pre>void Economic() { GIFTS I; ifstream fin("GIFTS.DAT",ios::binary); while (fin.read((char *)&I,sizeof(I))) { if(I.GetCost(>2000) I.See(); } fin.close(); }</pre> <p>(1 Mark for opening GIFTS.DAT correctly) (1 Mark for reading all records from the file) (1 Mark for checking value of Cost > 2000) (1 Mark for displaying the desired items and closing the file)</p>	4
(b)	<p style="text-align: center;">OR</p> <pre>void Read_File() { BUS B; ifstream Fin; Fin.open("Bus.Dat", ios::binary); while(Fin.read((char *) &B, sizeof(B)))</pre>	

	<pre> { if(strcmp(B.EndTo(), "Mumbai")==0) { B.show() ; } Fin.close(); } </pre> <p>(1 Mark for opening Bus.dat correctly) (1 Mark for reading all records from the file) (1 Mark for checking destination is Mumbai) (1 Mark for displaying the desired items and closing the file)</p>	
(c)	<p>(i) File.seekg(-1 * sizeof(I) ,ios: :cur)); (1 Mark) (ii) File.write((char*)&I,sizeof(I)); (1 Mark)</p>	2
(c)	<p style="text-align: center;">OR</p> <p>tellp(): This function returns the position of the current put pointer in terms of bytes in a file. int n = f.tellp(); (1 Mark) seekp(): This function takes the file put pointer to the specified byte in a file. Eg: f.seekp(30); // It takes a pointer to 30th byte. (1 Mark)</p>	
8(a)	<p>(½ Mark for function header) (1 Marks for the correct Logic to search an integer using binary search) (½ Mark for correct return statement)</p>	2
(b)	<p>(½ Mark for function header) (1 Mark for correct loop) (1½ Marks for the correct Logic for sorting)</p>	3
(c)	<p>(½ Mark for function header) (2½ Marks for the correct Logic)</p>	3
(d)	<pre> void DISPMID(int A[][5],int R,int C) { for (int J=0;J<C;J++) cout<<A[R/2][J]<< " "; cout<<endl; for (int I=0;I<R;I++) cout<<A[I][C/2]<< " "; } </pre> <p>(½ Mark for function header) (2 Marks for the correct Logic to display middle row and middle column)</p>	3
(d)	<p style="text-align: center;">OR</p> <pre> void SWAPCOL(int A[][100], int M, int N) { int Temp, I; for(I=0; I<M; I++) { Temp = A [I][0] ; A[I][0] = A[I][N-1]; A[I][N-1] = Temp; } } </pre>	

	}} 1 Mark for correct loop) (2 Mark for swapping the first column with last column correctly)	
(e)	$\text{Loc}(T[I][J]) = \text{BaseAddress} + W [(I - \text{LBR}) * C + (J - \text{LBC})]$ (where $W = \text{size of each element} = 4 \text{ bytes}$, $R = \text{Number of Rows} = 20$, $C = \text{Number of Columns} = 50$) Assuming $\text{LBR} = \text{LBC} = 0$ $\text{LOC}(T[10][8]) \quad 52000 = \text{BaseAddress} + W [I * C + J]$ $52000 = \text{BaseAddress} + 4[10 * 50 + 8] \quad 52000 = \text{BaseAddress} + 4[500 + 8]$ $52000 = \text{BaseAddress} + 4 \times 508$ $\text{BaseAddress} = 52000 - 2032 = \mathbf{49968}$ $\text{LOC}(T[15][5]) = \text{BaseAddress} + W [I * C + J] = 49968 + 4[15 * 50 + 5]$ $= 49968 + 4[750 + 5]$ $= 49968 + 4 \times 755$ $= 49968 + 3020$ $= \mathbf{52988}$ 1 Mark for writing correct formula OR substituting formula with correct values) (1 Mark for correct step calculations) (1 Mark for final correct address)	3
(e)	<p style="text-align: center;">OR</p> Address of Array[i][j] along the column $= \text{Base Address} + W [(i - L1) + (j - L2) * M]$ where, $W = \text{size of each location in bytes} = 4$ $L1 = \text{Lower Bound of rows} = 0$ $L2 = \text{Lower Bound of columns} = 0$ $M = \text{Number of rows per column} = 40$ Address of Array[30][10] = Base Address + $4 * (30 + 10 * 40)$ $9000 = \text{Base Address} + 4 * 430$ $\text{Base Address} = 9000 - 4 \times 430 = 9000 - 1720 =$ $\mathbf{7280}$ Address of Array[3][6] = $7280 + 4 * (3 + 6 * 40) = 7280 + 4 * 243 = 7280 + 972 =$ $\mathbf{8252}$ 1 Mark for writing correct formula OR substituting formula with correct values) (1 Mark for correct step calculations) (1 Mark for final correct address)	
9(a)	(½ Mark for checking function header) (2½ Mark for logic to add a book information in stack)	3
(b)	(½ Mark for checking function header) (2½ Mark for logic deleting the product from the Queue)	3
(b)	<p style="text-align: center;">OR</p> (½ Mark for checking function header) (2½ Mark for logic to add a member in the Queue)	

(c)	<p>Operator Scanned Stack Content</p> <table border="1" data-bbox="207 170 1208 552"> <thead> <tr> <th>Operator scanned</th><th>Stack Content</th><th>Intermediate Output</th></tr> </thead> <tbody> <tr><td>4</td><td>4</td><td></td></tr> <tr><td>10</td><td>4,10</td><td></td></tr> <tr><td>5</td><td>4,10,5</td><td></td></tr> <tr><td>+</td><td>4,15</td><td>10+5=15</td></tr> <tr><td>*</td><td>60</td><td>4*15=60</td></tr> <tr><td>15</td><td>60,15</td><td></td></tr> <tr><td>3</td><td>60,15,3</td><td>15/3=5</td></tr> <tr><td>/</td><td>60,5</td><td>60-5=55</td></tr> <tr><td>-</td><td>55</td><td></td></tr> </tbody> </table> <p>Result = 55 1 Marks for correct steps showing stack status) (1 Mark for correct output)</p>	Operator scanned	Stack Content	Intermediate Output	4	4		10	4,10		5	4,10,5		+	4,15	10+5=15	*	60	4*15=60	15	60,15		3	60,15,3	15/3=5	/	60,5	60-5=55	-	55		2
Operator scanned	Stack Content	Intermediate Output																														
4	4																															
10	4,10																															
5	4,10,5																															
+	4,15	10+5=15																														
*	60	4*15=60																														
15	60,15																															
3	60,15,3	15/3=5																														
/	60,5	60-5=55																														
-	55																															
(d)	<p>Element Scanned Stack Postfix $X - Y / (Z + U) * v$ $= (X - ((Y / (Z + U)) * v))$</p> <p>Postfix expression= XYZU+/V*- (1 Mark for correctly converting till each operator) (1 Mark to be given for writing correct answer)</p>	2																														