INDIAN SCHOOL MUSCAT

FIRST TERM EXAMINATION

SEPTEMBER 2018

CLASS XI

Marking Scheme – COMPUTER SCIENCE [THEORY]

SET B

Q.NO.	Answers	Marks
1a.	**CISC-** complex instruction set computer refers to computers designed with a full set of computer instructions that were intended to provide needed capabilities in the most efficient way**. -½ mark** **RISC**- Reduced instruction set computer – is a microprocessor that is designed to perform a smaller number of types of computer instructions so that it can operate at a higher speed MIPS. Each instruction type that a computer must perform require additional transistors/circuit, a large list of instructions tends to make microprocessor more complicated and slow. **- ½ mark**	1
b.	**Cache memory** is a high-speed storage mechanism that stores the most recently accessed data. **-½ mark** It makes CPU run faster if the required data is found in cache. **- ½ mark**	1
c.	**SRAM-** the information stored in static RAM need not be refreshed but it remains stable as long as the power supply is provided. SRAM is costlier but has higher speed than DRAM. **– ½ mark** **DRAM** - the information stored in Dynamic RAM need to be refreshed after every few milliseconds. Otherwise it will be erased. DRAM has higher storage capacity and is cheaper than static RAM. **– ½ mark**	1
d.	Any 2 characteristics - **- ½ mark each**	1
e.	**Serial Port**: Transfers data serially a bit at a time, as a result it needs only wire to transmit 8 bits. , it takes longer to transmit a byte.It comes in the form of 9 pin or 25 pin male connector. Used to connect devices mouse , modem. **- 1 mark**	1
f.	Explanation **of Bluetooth - 1 mark**	1
g.	i) USB Port - Explanation **-1 mark** ii) AGP Port - Explanation **-1 mark**	2
h.	i) note on Blu Ray Disc **- 1 mark** ii) note on CD's **- 1 mark**	2
2a.	main() is the point where C++ begin its execution and ends its execution with it. **– 1 mark**	1
b.	There are two ways to insert comments – i) Single line comment- the comments that begin with //- Explanation **- 1 mark** ii) Multiline or block comments- the block comments mark the beginning with /* and ends with */ - Explanation **- 1 mark**	2
c.	i) **Token** - The smallest individual unit in a program. e.g. keywords(any)**– 1 mark** ii) **Runtime** error is that occurs during the execution of the program. Example – divide a number by zero. **– 1 mark**	2
3a.	**Dynamic initialization**- A variable can be initialized at run time using expressions and this way of initializing is known as dynamic initialization. – **½ mark** e.g. float avg=total/n; **-½ mark**.	1
b.	**Access modifier const**- modifies the access type of a variable.	2

	const int val=10 The value of val cannot be changed through out the program. explanation - **1 mark** example – **1 mark**	

c.	Advantages over integers – **1 mark**
	    i) Can represent values between two integers
	    ii) Can represent a greater range of values
	Disadvantage (upto 7 decimal places) – **1 mark**
	    i) Floating point operations are slower than integer operations.
		2

4a.	**Implicit type conversion**- Conversion performed by the compiler whenever differing data types are inter-mixed in an expression. – **½ mark** **Explicit Type Conversion**-(Type casting) Conversion of an operand to a specific type explicitly by the user in the program is known as Type casting. – **½ mark** e.g. (int)(a+b/2)	1
b.	(a>b)?c=a:c=b; – **1 mark**	1
c.	= Assignment operator –Explanation - **½ mark** example - – **½ mark** == Relational operator– Explanation -**½ mark** example - – **½ mark**	2
d.	i) ch>=65 && ch<=90 – **1 mark** II) weight >=115 && weight <125 – **1 mark**	2

e.	**i) 12 7 – ½ mark for each**
	  ii) S=71 – ½ mark
	    T=81 – ½ mark
		2

f.	**i) fabs((exp(x)-cos(x)) – 1 mark** **ii) pow(x,6)+pow(y,3)-atan(x)/y - 1 mark**	2
5a.	**fall-through** – If a case statement does not include a break statement then the control continues right on the next case until either break is encountered or end of switch is reached. This situation is called "fall-through" – **½ mark** **significance**-default statement gets executed when no match is found. – **½ mark**	1
b.	**Empty loop** -If we put semicolon after for's parenthesis it repeats only for counting control variable. – **½ mark** Example: for(i=1;i<=10;++i); **-½ mark**	1
c.	for loop and while loop, are **entry-controlled loops**, the test expression is evaluated at the beginning of the loop before executing the loop body. **– 1 mark** do-while loops are **exit -controlled loop** , it evaluates test expression at the bottom of the loop after executing its loop body statements. This means do while loop always executes at least once. **– 1 mark**	2
d.	int p=25; - **½ mark** do {cout<<p<<"\t"; p+=25; - **½ mark** } while(p<=500); - **½ mark** cout<<"end of loop"; - **½ mark**	2
e.	ABCDEFGH ABCDEF ABCD AB - **½ mark** *4	2
f.	#include<iostream.h>	2

	void main**()** - **½ mark** - **½ mark**
	{int i=1 **,** s=1; - **½ mark**
	for(;i<5;s*=i,++i)
	 cout**<<**s; - **½ mark**
	}	

g.	char code;
	
	  cin >> code;
	
	  switch(code)
	
	  {
	
	  case 'S':
	
	       cout<<"Summer ";
	
	  case 'W':
	
	       cout<<" Winter ";
	
	  case 'R':
	
	       cout<<" Rainy ";
	
	  default:
	
	       cout<<"Enter correct code";
	
	  }
	
	Proper use of switch - ½ mark case -1 mark default- ½ mark brackets - ½ mark
		2

6.a.	for Declaration & input statements **- 1 mark** for correct logic to find the largest and displaying **- 2 marks**	3
b.	Header file, declarations **-1 mark** Correct nested loop & output statement with endl at correct place **-2 marks**	3
c.	for Declaration & input statements **- 1 mark** for correct logic and displaying the series **- 2 marks**	3
d.	for Declaration & input statements **- 1 mark** for correct logic and displaying the Multiplication table **- 2 marks**	3
7a.	for declaration & input statements **-1 mark** for correct logic to find the Income Tax **-2½ marks** for displaying - **½ mark**	4
b.	menu - **1 mark** correct logic for choice 1 - **1 mark** correct logic for choice 2 - **1 mark** correct logic for choice 3 and continuation - **1 mark**	4
c.	for declaration & input statements **-1 mark** for correct logic to check prime number **-2½ marks** for displaying - **½ mark**	4
d.	for declaration & input statements **-1 mark** for correct logic to find the sum of the series **-2½ marks** for displaying - **½ mark**	4