KENDRIYA VIDYALAYA SANGATHAN

KOLKATA REGION

CLASS: XI

SESSION: 2017 - 18

SPLIT-UP SYLLABUS

COMPUTER SCIENCE - THEORY

MONTH	PORTION TO BE COVERED	Periods	
		THEORY	PRACTICAL
JUNE-JULY	**UNIT 1: COMPUTER FUNDAMENTALS** **Classification of computers:** Basics of computer and its operation; Functional Components andtheir interconnections, concept of Booting. **Software concepts:** Types of Software – System Software, Utility Software and ApplicationSoftware **System Software:** Operating System, Compiler, Interpreter and Assembler; **Operating System**: Need for Operating System, Functions of Operating System (ProcessorManagement, Memory Management, File Management and Device Management), Types ofOperating System-Interactive (GUI based), Time Sharing, Real Time and Distributed, Commonlyused Operating System: UNIX, LINUX, Windows, Solaris, BOSS (Bharat Operating System Solutions);Mobile OS – Android, Symbian, IOS. **Utility Software:** Anti Virus, File Management tools, Compression tools and Disk Management tools(Disk Cleanup, Disk Defragmenter, Backup). **Open Source Concepts:** Open Source Software, Freeware, Shareware, and Proprietary Software. **Application Software:** Office Tools – Word Processor, Presentation Tool, Spreadsheet Package,Database Management System; Domain Specific tools – School Management System, InventoryManagement System, Payroll System, Financial Accounting, Hotel Management, ReservationSystem and Weather	18	6

	Forecasting System. **Number System:** Binary, Octal, Decimal, Hexadecimal and conversion between different numbersystems. **Internal Storage encoding of Characters:** ASCII, ISCII (Indian Scripts Standard Code forInformation Interchange), and UNICODE (for multilingual computing) **Microprocessor:** Basic concepts, Clock speed (MHz, GHz), 16 bit, 32 bit, 64 bit, 128 bitprocessors; Types – CISC Processors (Complex Instruction Set Computing), RISC Processors(Reduced Instruction Set Computing), and EPIC (Explicitly Parallel Instruction Computing). **Memory Concepts:** Units: Byte, Kilo Byte, Mega Byte, Giga Byte, Tera Byte, Peta Byte, Exa Byte,Zetta Byte, Yotta Byte. **Primary Memory**: Cache, RAM, ROM **Secondary Memory**: Fixed and Removable storage – Hard Disk Drive, CD/DVD Drive, Pen Drive,Blue Ray Disk. **Input Output Ports/ Connections:** Serial, Parallel and Universal Serial Bus, PS-2 port, Infraredport, Bluetooth, Firewire.		
PERIODIC TEST – I **PORTION:** **1.** Computer overview [14] **2.** Working with operating system [16] **3.** Data representation [04] **4.** Microprocessor basics [06] **MAXIMUM MARKS: [14 + 16 + 4 + 6 = 40 Marks]** **DURATION: 90 Minutes**			
AUG	**<u>UNIT 2: PROGRAMMING METHODOLOGY</u>** **General Concepts:** Modular Approach, Clarity and Simplicity of Expressions, Use of proper namesfor Identifiers, Comments, Indentation; Documentation and Program Maintenance; Running andDebugging programs, Syntax Errors, Run-Time Errors, Logical Errors **Problem Solving Methodologies:** Understanding of the problem,	28	10

	Solution for the problem,Identifying minimum number of inputs required for output, Writing code to optimizing executiontime and memory storage, step by step solution for the problem, breaking down solution intosimple steps (modular approach), Identification of arithmetic and logical operations required forsolution; Control Structure- Conditional control and looping (finite and infinite). **Problem Solving:** Introduction to Algorithms/Flowcharts. **UNIT-3: INTRODUCTION TO C++** **Getting Started:** C++ character set, C++ Tokens (Identifiers, Keywords, Constants, Operators,),Structure of a C++ Program (include files, main function), Header files – iostream.h, iomanip.h,**cout, cin;** use of I/O operators (<<and>>), Use of endl and setw ( ), Cascading of I/O operators,compilation , Error Messages; Use of editor, basic commands of editor, compilation, linking andexecution. **Data Types, Variables and Constants:** Concept of Data types; Built-in Data types: **char, int , float**and **double;** Constants: Integer Constants, Character constants (- \n, \t, \b), Floating PointConstants, String Constants; Access modifier: **const;** Variables of built-in-data types,Declaration/Initialization of variables, Assignment statement, Type modifier: **signed, unsigned,long** **Operator and Expressions:** Operators: Arithmetic operators (-,+,*,/,%), Assignment operator (=),c++shorthands (+=, -=,*=,/=,%=) Unary operators (-), Increment (++) and Decrement (--)Operators, Relational operator (>,>=,<=,=,!=), Logical operators (!,&&,||), Conditional operator:<condition>?<if—true>:<if false>; Precedence of Operators; Automatic type conversion inexpressions, Type casting;	44	36
SEP	**UNIT 4: PROGRAMMING IN C++** **Flow of control** **Conditional statements**: **if else,** Nested **if, switch..case..default,** use of conditional operator,Nested **switch..case, break** statement		

<table>
<tr><td></td><td>(to be used in switch..case only); Loops: while, do –while, for and Nested loops.</td><td></td><td></td></tr>
<tr><td><u>OCT</u></td><td><u>FUNCTIONS</u>
Inbuilt Functions

<table>
<tr><th>Header file Categorization</th><th>Header File</th><th>Function</th></tr>
<tr><td>Standard input/output functions</td><td>stdio.h</td><td>gets ( ), puts ( )</td></tr>
<tr><td>Character Functions</td><td>ctype.h</td><td>isalnum ( ), isalpha ( ),
isdigit ( ), islower ( ),
isupper ( ), tolower ( ),
toupper ( )</td></tr>
<tr><td>String Function</td><td>string.h</td><td>strcpy ( ), strcat ( ),
strlen ( ), strcmp ( ),
strcmpi ( ), strrev ( ),
strupr ( ), strlwr ( )</td></tr>
<tr><td>Mathematical Functions</td><td>math.h</td><td>fabs ( ), pow ( ), sqrt ( ), sin ( ),
cos ( ), abs ( )</td></tr>
</table>

Introduction to user-defined function and its requirements.
Defining a function; function prototype, Invoking/calling a function, passing arguments tofunction, specifying argument data types, default argument, constant argument, call by value,call by reference, returning values from a function, calling functions with arrays, scope rules ofvariables: local and global variables.Relating to Parameters and return type concepts in built-in functions.</td><td>20

6</td><td>18

2</td></tr>
<tr><td><u>NOV</u></td><td><u>STRUCTURED DATA TYPE</u>
Arrays: Introduction to Array and its advantages.
One Dimensional Array: Declaration/initialization of One-dimensional array, Accepting arrayelements, accessing array elements, manipulation of array elements (sum of elements, product ofelements, average of elements, linear search, finding maximum/minimum value), Declaration / Initialization of a String, string manipulations (counting vowels/ consonants/ digits/ special characters, case conversion, reversing a string, reversing each word of a string)</td><td>14</td><td>18</td></tr>
<tr><td colspan="4">HALF YEARLY EXAMINATION
PORTION:
1. Computer Fundamentals (Portion Covered in Periodic Test - I) [10]
2. Programming Methodology [12]
3. Introduction to C++:Getting Started [08]</td></tr>
</table>

4. Data Types, Variables and Constants [10]
5. Operator and Expressions [10]
6. Flow of control [10]
7. Functions [10]

MAXIMUM MARKS: [10 + 12 + 08 + 10 + 10 + 10 + 10 = 70 Marks]
DURATION: 03 Hours

DEC	**Two-dimensional Array:** Declaration/initialization of a two-dimensional array, inputting arrayelements, accessing array elements, manipulation of Array elements (sum of row element, columnelements, diagonal elements, finding maximum / minimum values) **User-defined Data Types:** Introduction to user defined data types.	16	3
JAN	**Structure:** Defining a Structure (Keyword Structure), declaring structure variables, accessingstructure elements, passing structure to functions as value and reference, argument/parameter,function returning structure, array of structure, passing an array of structure as an argument/ aparameter to a function.Defining a symbol name using **typedef**keyword and defining a macro using **#define preprocessor**directive.		

PERIODIC TEST – II
PORTION

1. Library Functions [06]
2. User defined functions [06]
3. Array – Single Dimension [16]
4. Array – Two Dimension [12]

MAXIMUM MARKS: [6 + 6 + 16 + 12 = 40 Marks]
DURATION: 90 Minutes

FEB	**Revision and Project Work**		
MAR	**SESSION ENDING EXAMINATION** **PORTION : WHOLE SYLLABUS**		

Duration: 3 hours | Total Marks: 70

Unit No.	Unit Name	Marks
1.	COMPUTER FUNDAMENTALS	10
2.	PROGRAMMING METHODOLOGY	12
3.	INTRODUCTION TO C++	14
4.	PROGRAMMING IN C++	34
	Total	70

SPLIT-UP SYLLABUS
COMPUTER SCIENCE – PRACTICAL

MONTH	PORTION TO BE COVERED	PRACTICAL PERIODS
JUNE-JULY	**UNIT 1: COMPUTER FUNDAMENTALS** **1.** Opearting System Practicals (Commands, CUI, GUI, Hands on Experience) **2.** Packages and Utility Softwares **3.** Record of the configuration of computer system used by the student in the computer lab (by exploring inside computer system in the first 2 lab classes).	6
AUG	**PROGRAMMING METHODOLOGY** Use of comments and indentation, formatting styles in C++, Documentation, Algorithms and Flowcharts, Errors correction and Debugging, Program Maintenance. **INTRODUCTION TO C++** Structure of a C++ Program (include files, main function), Header files – iostream.h, iomanip.h, **cout, cin statements;** use of I/O operators (<<and>>), Use of endl and setw ( ), Cascading of I/O operators,Error Messages, Use of editor, basic commands of editor, compilation, linking and execution. **Data Types, Variables and Constants:** Built-in Data types: **char, int , float** and **double;** Constants: Integer Constants, Character constants (- \n, \t, \b), Floating Point Constants, String Constants; Access modifier: **const;** Variables of built-in-data types,	10 36

<table>
<tr>
<td></td>
<td>Declaration/Initialization of variables, Assignment statement, Type modifier: signed, unsigned, long
Operator and Expressions: Operators: Arithmetic operators (-,+,*,/,%), Assignment operator (=), c++shorthands (+=, -=,*=,/=,%=) Unary operators (-), Increment (++) and Decrement (--) Operators, Relational operator (>,>=,<=,=,!=), Logical operators (!,&&,||), Conditional operator: <condition>?<if—true>:<if false>;</td>
<td></td>
</tr>
<tr>
<td>SEP</td>
<td><u>UNIT 4: PROGRAMMING IN C++</u>
Flow of control
Conditional statements: if else, Nested if, switch..case..default, use of conditional operator,Nested switch..case, break statement (to be used in switch..case only); Loops: while, do –while, for and Nested loops , jump statements.
Minimum 05 Programs on Control structures</td>
<td>48</td>
</tr>
<tr>
<td><u>OCT</u></td>
<td><u>FUNCTIONS</u>
Inbuilt Functions
<table>
<tr><th>Header file Categorization</th><th>Header File</th><th>Function</th></tr>
<tr><td>Standard input/output functions</td><td>stdio.h</td><td>gets ( ), puts ( )</td></tr>
<tr><td>Character Functions</td><td>ctype.h</td><td>isalnum ( ), isalpha ( ),
isdigit ( ), islower ( ),
isupper ( ), tolower ( ),
toupper ( )</td></tr>
<tr><td>String Function</td><td>string.h</td><td>strcpy ( ), strcat ( ),
strlen ( ), strcmp ( ),
strcmpi ( ), strrev ( ),
strupr ( ), strlwr ( )</td></tr>
<tr><td>Mathematical Functions</td><td>math.h</td><td>fabs ( ), pow ( ), sqrt ( ), sin ( ),
cos ( ), abs ( )</td></tr>
</table>
Introduction to user-defined function and its requirements.
Defining a function; function prototype, Invoking/calling a function, passing arguments tofunction, specifying argument data types, default argument, constant argument, call by value,call by reference, returning values from a function, calling functions with arrays, scope rules ofvariables: local and global variables.Relating to Parameters and return type concepts in built-in functions.</td>
<td>20</td>
</tr>
</table>

NOV	**STRUCTURED DATA TYPE** **Arrays:** Introduction to Array and its advantages. **One Dimensional Array**: Declaration/initialization of One-dimensional array, Accepting arrayelements, accessing array elements, manipulation of array elements (sum of elements, product ofelements, average of elements, linear search, finding maximum/minimum value), Declaration / Initialization of a String, string manipulations (counting vowels/ consonants/ digits/ special characters, case conversion, reversing a string, reversing each word of a string) Minimum 05 Programs on String manipulations	18
DEC	**Two-dimensional Array:** Declaration/initialization of a two-dimensional array, inputting arrayelements, accessing array elements, manipulation of Array elements (sum of row element, columnelements, diagonal elements, finding maximum / minimum values) **User-defined Data Types:** Introduction to user defined data types. Minimum 10 Programs on array manipulations(1D & 2D)	3
JAN	**Structure:** Defining a Structure (Keyword Structure), declaring structure variables, accessingstructure elements, passing structure to functions as value and reference, argument/parameter,function returning structure, array of structure, passing an array of structure as an argument/ aparameter to a function.Defining a symbol name using **typedef**keyword and defining a macro using **#define preprocessor**directive. Minimum 05 Programs on structures.	
FEB	**Revision and Project Work** Problems using String, Number, array and structure manipulation General Guidelines: Initial Requirement, developing an interface for user (it is advised to use text basedinterface screen), developing logic for playing the game and developing logic for scoring points. • Memory game: A number guessing game with application of 2 dimensional arrays containingrandomly generated numbers in	

<table>
<tr><td></td><td>pairs hidden inside boxes.
・ Hollywood/Hangman: A word Guessing game
・ Cows 'N Bulls: A word/number Guessing game
・ Random Number Guessing Game (High\Low)
・ A game to check whether a word does not use any of the forbidden letters
・ Cross ‘N knots game: A regular tic-tac –toe game.
or
Similar projects may be undertaken in other domains. (As mentioned in general guidelines forproject, given at the end of the curriculum in a group of 2-4 students)
Collaboration and Presentation of the project</td><td></td></tr>
<tr><td><u>MAR</u></td><td>SESSION ENDING EXAMINATION
PORTION : WHOLE SYLLABUS
Duration: 3 hours Total Marks: 70
<table><tr><th>Unit No.</th><th>Unit Name</th><th>Marks</th></tr><tr><td>1.</td><td>COMPUTER FUNDAMENTALS</td><td>10</td></tr><tr><td>2.</td><td>PROGRAMMING METHODOLOGY</td><td>12</td></tr><tr><td>3.</td><td>INTRODUCTION TO C++</td><td>14</td></tr><tr><td>4.</td><td>PROGRAMMING IN C++</td><td>34</td></tr><tr><td></td><td>Total</td><td>70</td></tr></table></td><td></td></tr>
</table>

Class XI (Practical) – C++

Duration: 3 hours **Total Marks: 30**

1. Programming in C++ **10**

One programming problem in C++ to be developed and tested on Computer during theexamination.Marks are allotted on the basis of following:
Logic : 6 Marks
Documentation : 2 Marks
Output presentation : 2 Marks

2. Project Work **6 + 4**

Problems using String, Number, array and structure manipulation

General Guidelines: Initial Requirement, developing an interface for user (it is advised to use text basedinterface screen), developing logic for playing the game and developing logic for scoring points.

- Memory game: A number guessing game with application of 2 dimensional arrays containingrandomly generated numbers in pairs hidden inside boxes.
- Hollywood/Hangman: A word Guessing game
- Cows 'N Bulls: A word/number Guessing game
- Random Number Guessing Game (High\Low)
- A game to check whether a word does not use any of the forbidden letters

- Cross'N knots game: A regular tic-tac –toe game.

or

Similar projects may be undertaken in other domains. (As mentioned in general guidelines for project, given at the end of the curriculum in a group of 2-4 students)

-- Collaboration and Presentation of the project

3. Practical File **5+1**

(a) Record of the configuration of computer system used by the student in the computer lab (by exploring inside computer system in the first 2 lab classes).

(b) Must have minimum 20 programs from the topics covered in class XI course.

- Programs on Control structures
- Programs on String manipulations
- Programs on array manipulations(1D & 2D)
- Programs on structures.

*****1 mark is for innovating while developing programmes.

4. Viva Voce **4**

Viva will be asked from the syllabus covered in class XI and the project developed by the student(s).