

GOA BOARD OF SECONDARY AND HIGHER SECONDARY EDUCATIONAL  
ALTO-BETIM, GOA.

MODEL QUESTION PAPER FOR PUBLIC EXAMINATION

CLASS : XII

STREAM : GENERAL

SUBJECT : COMPUTER SCIENCE

Duration : 2½ Hours

Maximum Marks : 55

Instructions:-

- 1) This Question paper consists of 5 main questions.
- 2) All questions are compulsory.
- 3) Use of calculators & log tables **NOT** allowed.
- 3) Figures to the right indicate full marks.
- 4) State your assumptions wherever necessary.
- 5) All program to be written in C++ only.

Q1] A] Write the correct alternative from those given below: [01]

From a four base classes one class is derived, State the type of interitance

- A] Single      B] Multiple      C] Multilevel      D] Hierarchical

B] What is the purpose of seekg()? [01]

C] Evaluate the following postfix experssion and show the contents of the stack at the end of each operation. [02]

10, 20, +, 10, 5, -, 2, 4, \*, %, /

D] Consider the following class declaration & answer the question given below:

```
class BANK
{
    int bcode;
    protected:
        int capital();
    public:
        void readdata();
        void showdata();
};
class BRANCH:protected BANK
{
    int branch_no;
    protected:
        char location[15];
        int noofstaff;
    public:
        void getdata();
        void putdata();
};
class CUSTOMER: public BRANCH
{
    char name[20];
    int accno;
    protected:
        float deposit();
        float withdraw();
    public:
        float balance;
        void putdata();
};
```

i) Declare a default constructor for the base class BANK.

- ii) List the protected members of class CUSTOMER.
- iii) List the name of nthe members accessible from the objects of the class CUSTOMER.
- iv) What type of inheritance is used in the above declaration?.

E] Write any four characteristic of a construtor. [02]

F] Assume that a text file called “info.txt” contains some lower-case text written into it. Write a function in C++ called count(), that reads the file “info.txt” line by line and display numbers of vowels present on each line. [03]

Q2] A] Write the correct alternative from those given below: [01]

An array initially contains the following numbers

7, 5, 2, 3, 11, 3, 17

and after third pass contains the following

2, 3, 3, 5, 11,7, 17

which sorting algorithm is applied?

- a) Bubble sort   b) Insertion sort   c) Selection sort   d) Merge sort

B] What happens when an existing file is opened in ios::out mode? [01]

C] Consider the following class declaration

```
class Q
{
    struct node
    {
        int rno;
        char name[25];
        node *next;
    }*front,*rear;
public:
    Q(){front=rear=NULL;}
    void addq();           //to add an element to a queue
    void delq();           //to delete an element from a queue
    void display();
};
```

Write function definition for addq().

D] Consider the following circuit diagram & answer the question given below:

A

B

C

- i) Obtain Boolean expression for the output labeled F.
- ii) Simplify F and draw the simplified logic circuit diagram. [02]

E] Illustrate with the help of a diagram as to how a 3-input OR gate can be designed using 2-input NOR gates only. [02]

F] Give one point of difference between each of the following:

- i) Textfile and Binary file.
- ii) ios: : app & IOS : :ate
- ii) get pointer & put pointer [03]

Q3] A] Write the correct alternative from those given below: [01]  
Which of the boolean statement is incorrect?  
a)  $A + A' = 1$     b)  $A.A' = 1$     c)  $A + 1 = 1$     d)  $A.1 = A$

B] What is a URL? [01]

C] With the help of neatly labeled diagram describe a Coaxial cable [02]

D] Briefly explain the role of the following devices with respect to networking: [02]  
i) repeater  
ii) router

E] Write a short note on satellite communication. [02]

F] Write a function in C++ to display all three Armstrong numbers. [03]  
[OR]

F] Write a function in C++ to display the first 'n' prime numbers. [03]

Q4] A] Write the correct alternative from those given below: [01]  
Which of the following operator is evaluated first?  
a) \*    b) +    c) <=    d) &&

B] Rewrite the following IF-ELSE statement in terms of SWITCH-CASE statement: [01]  

```
int innum, calval;
cin>>innum;
if (innum==5)
    count<<innum+calval;
else if (innum==10)
    count<<innum-calval;
```

C] Write a short note on Public visibility label. [02]

D] Consider the unsorted array A

A 80 94 91 12 45 77

The above array is sorted using some sorting algorithm and the array contents at the end of each pass are shown below

Pass 1 94 80 91 12 45 77

Pass 2 94 91 80 12 45 77

Pass 3 94 91 80 12 45 77

Pass 4 94 80 91 45 12 77

Pass 5 94 80 91 77 45 12

Identify the sorting algorithm applied to sort the above array and also write a function in C++ to accept an array & its size and sort it in the similar manner as shown in the example given above. [03]

E] Write a C++ program with the following class specifications.  
Declare an abstract class called "COLLEGE" with two private data members Cname(college name) & place(location of the college). Also define a parameterised constructor which only initialises the data members and a public member function called display() to display the data members.

Declare another abstract class called "COURSE" with one private data member called cou\_name(course name). Define a parameterised constructor to initialise the data

members and a public member function called display() to display the data members.

Declare a class called "STUDENT" privately from COLLEGE & COURSE consisting of one private data member called sname(student's name), a parameterised constructor to initialise the data members and a public member function called display() to display the data members.

Write main() function to declare object(s) and call the necessary function to obtain the output in the following format. [04]

Student's name : Ms.Vedika Arya  
Course : Mechanical Engineering  
College : GEC  
Place : Ponda

[OR]

E] Consider the following class declaration :

```
class goods
{
    int code;
    float price;
    char gname[15];
public;
    -----
};
class shop
{
    char name[15];
    char owner[15];
public;
    -----
};
```

Complete the above two classes and derive a class 'customer' containing the data members to store the customers name and city from the above 2 classes. Also write main() to store the following data in the object of class customer type:

Customer=> name : Vedika Chawla, City : Ponda

Goods => Code : XT345, Price : 350, gname : wall clock

Shop => Name : Paradise Stationeries, owner : Vishnu Mittal

[04]

Q5] A] Write the correct alternative from those given below:

[01]

Consider the following program segment :

```
int i=6720, j=4;
while( ( i % j ) == 0)
{
    i = i / j;
    j += 1;
}
```

on termination j will have a value \_\_\_\_\_

- a) 4      b) 8      c) 9      d) 6720

B] What is polymorphism? [01]

C] What are inline functions? Under which circumstances will a function not execute as an inline function [02]

D] Simplify the following Boolean expression using K-map & also draw the logic circuit diagram for the simplified expression using NAND gates only.

$$F(p,q,r,s)=m_0 + m_1 + m_2 + m_3 + m_4 + m_6 + m_9 + m_{11} \quad [03]$$

E] Consider the following class declaration

```
class LL
{
    struct node
    {
        int ino;    //item code
        float price; //item price
        node *link;
    }*start;
public:
    LL(){start=NULL;}
    void create():
    void display():
    void remove():
};
```

Write a function definition for remove(), to remove a node from the linked list containing the item code 'x' in the data part of a node.

If no such node exists the display appropriate message. [04]

E] Consider the following class declaration

```
class LIST
{
    struct NODE
    {
        float data;
        NODE *next;
    }*START;
};
```

Add a member function to the above class called INSERT() to insert a new node at a location LOC ( $1 \leq LOC \leq N$ ) to the linked list. Assume that list already contains 'N' nodes. [04]

**COMPUTER SCIENCE SYLLABUS**  
**STD XII**

UNIT NO	NAME OF TOPIC	MAX MARKS
1	PROGRAMMING IN C++	20
2	DATA STRUCTURES USING C++	12
3	FILE HANDLING	08
4	BOOLEAN ALGEBRA	08
5	COMPUTER NETWORKS	07
TOTAL		55

**UNIT-I : PROGRAMMING IN C++ (20 MARKS)**

**1. C++ Fundamentals:**

- C++ character set
- Identifiers and Keywords
- Data Types: int, float, char, double, void
- Qualifiers: short, long, signed, unsigned, const
- Constants (Integer, Floating point, character, string, enumeration constants, symbolic constants)
- Escape Sequence (\b, \t, \n, \v, \a, \f, \r, \0)
- Variables and Declaration, Dynamic initialization of variables, reference Variables

**2. Operators and Expressions:**

- Unary Operators: unary minus, ++, --, !, sizeof(), typeid
- Arithmetic Operators: \*, /, %, +, -
- Relational Operators: <, <=, >, >=
- Equality Operators: ==, !=
- Logical Operators: &&, ||
- Conditional Operator: ?:
- Assignment Operator: =, +=, -=, \*=, /=, %=
- Scope resolution operators (::)
- Memory management operators: new(), delete()
- Operator precedence and associativity

**3. Data Input and Output**

- Header file <iostream.h>
- Using cin and cout with Insertion and extraction operators
- Manipulators:

Definition, Header file <iomanip.h>

setw, endl, setprecision, setfill, setiosflags, resetiosflags

Flags : ios::left, ios::right, ios::scientific, ios::fixed, ios::showpos, ios::showpoint, ios::skipws, ios::unitbuf.

#### **4. Use of editor, basic commands of editor, Compilation, Linking and Execution of Program, Debugging.**

#### **5. Control Statements :**

If-else statement, while statement, do-while statement, for statement, switch statement.

break statement, continue statement

Comma operator.

#### **6. Functions:**

Definition, Concept, General Form, Function Declaration, Function Call (pass by value, pass by reference using pointers and pass by reference using reference variable), Function Definition , Calling Function with arrays as parameters, Return by Reference, Inline Functions, Functions with Default Parameters, Function overloading, Local and Global variables

**Built-in Functions:** <string.h>: strlen(), strcmp(), strcat(), strcpy()

<math.h> : log(), log10(), pow(), sqrt(), sin(), cos(), abs()

<ctype.h>: isalnum(), isdigit(), islower(), isupper(), tolower(), toupper(), isalpha(), isspace().

<stdio.h>: gets(), puts(), getchar(), putchar()

<conio.h>: clrscr(), getch()

#### **7. Basic concepts of Object Oriented Programming:**

Definition ,Objects, Classes, Data Abstraction, Data Encapsulation, Inheritance, Polymorphism

Characteristics of Object Oriented Programming

#### **8. Classes and Objects:**

Definition of class and object, Declaration of class, Defining member functions (inside the class and outside the class),Creating Objects, Accessing Class Members, Array of objects, Objects as function argument, Functions returning objects.

#### **9. Constructors and Destructors:**

Definition and characteristics of constructors, Default Constructor ,Constructor with Default Arguments, Parameterised constructors (explicit call, implicit call),Constructor overloading, Copy Constructor, Dynamic Constructor, Dynamic Initialization of objects, Destructor Definition and characteristics.

#### **10. Inheritance:**

Definition, Concept of Inheritance: Base , and derived classes, Type of Inheritance: single, multiple, multilevel, hierarchical and hybrid. Defining Derived Class: Visibility modes (public, private, protected),Public Derivation, Private Derivation, Protected Derivation, Virtual base classes, Abstract classes, Constructors in Derived classes, Containership.

## UNIT II : DATA STRUCTURES USING C++ (12 MARKS)

### 1) Arrays

- **One Dimensional arrays**

Definition, Declaration, Reading, Displaying, Accessing.

Algorithm and Program for inserting and deleting an element in an array. Memory allocation.

**Sorting:** Definition, sorting techniques (Insertion Sort, Selection Sort, Bubble Sort- algorithms and programs)

**Searching:** Definition, Searching Techniques (Linear Search, Binary Search- Algorithms and Programs)

**Merging:** Definition, Algorithm and Program to merge two sorted arrays

**Applications:** Insertion of an element in a sorted array, Displaying common elements of two single dimensional arrays and other applications.

- **Two Dimensional Arrays**

Definition, Declaration, Reading, Displaying, Accessing.

Applications: Matrix Addition, Transpose of a matrix, Matrix Multiplication, Representation of sparse matrix in 3- Tuple form, and other applications.

### 2) Structures and Pointers:

- **Pointers-** Definition, Concept, Declaration, Pointer to one and two dimensional array.

- **Structures-** Definition, Concept, Declaration, Structure variable, array of structures, pointer to a structure variable, pointer to array of structures.

### 3) Linked lists

- **Singly Linked list:**

Concept, Definition, Diagram, Operations (Creation, Display, Deletion of a node at any position, Insertion of a node at any position).

- **Circular Linked list:**

Concept, Definition, Diagram.

- **Doubly Linked List:**

Concept, Definition, Diagram

- Applications of singly linked list- Linear search and other applications.

- **Stacks:**

Concept, Definition, and implementation of a stack using linked list(PUSH,POP and display).

Applications of stacks (Infix, Postfix and Prefix Notations of expressions, Conversion of infix to postfix using stacks [Algorithms and problems only], Evaluation of postfix expressions[Algorithms and problems only]).

- **Queues:**

Linear queue- Concept, Definition, Implementation of queue using linked list( Add, Delete and Display).

Circular Queue- Concept, Definition.



### UNIT III: FILE HANDLING (08 MARKS)

- **Files:**  
Definition, Types of files-Text and Binary.
- **Stream Classes and their Member Functions.**  
Ifstream- get(),getline(),read(),seekg(),tellg(), Open(),close(),eof()  
Ofstream- put(),seekp().tellp(),write(),Open(),close().  
Fstream.
- **File Modes**-ios:: app, ios::ate, ios::in,ios::out,ios::binary, ios::trunc,ios::nocreate, ios::noreplace
- **Opening a file using constructor and using open member function.**
- **Closing a file.**
- **Detecting the end of a file.**
- **File Pointer and their manipulation.**
- **Text Files:**  
Creation, Display and File Processing (Character and String based processing)
  - **Binary Files:**  
Creation, Display and File Processing (Appending, Inserting, Deleting, Updating, Searching, Splitting and Merging)

### UNIT IV: BOOLEAN ALGEBRA (08 marks)

- **Basics of Boolean Algebra :-**  
Evolution of Boolean Algebra , Basic Terminology – Logical Statements, Logical Constants, Binary Valued Quantities , Compound Statements, Truth Table.
- **Logical operators:-**  
NOT, AND, OR , Switch, Switching Circuits (NOT, AND, OR).
  - **Postulates of Boolean algebra:-** Closure Property, Commutative Property , Associative Property  
Distributive Property , Identity Property , Inverse Property .
  - **Laws of Boolean Algebra :** Idempotent Law, Distributive Law, Absorption Law, Involution law.
  - **DeMorgan's Law and their applications.**
  - **Principle of Duality in Boolean algebra.**
  - **Derivation of Boolean expression:**
  - **Minterm , Maxterm , Shorthand Notation, Canonical Form, Sum of Product form (SOP), Product of Sum form (POS), Conversion of SOP to POS and vice versa, simplification of boolean expressions using postulates and laws of Boolean Algebra.**
  - **Karnaugh Maps:**  
Two variable K map, Three variable K map, Four variable K map, Pairing , Quads, Octet in K map, Simplification of K maps up to four variables , Overlapping groups , map rolling, eliminating redundant groups, use of K map for simplification and conversion of Boolean expression.
- **Logic gates:**  
Fundamental gates: AND gate, OR gate, NOT gate (Definition, Symbol, Truth table)
  - **Derived gates:**  
NOR gate, NAND gate, X-OR gate, X-NOR gate (Definition, Symbol, Truth Table), NAND and NOR gates as universal gates. Constructing logic circuits using basic gates and universal gates.
  - **Adder circuits:**  
Half Adder and Full Adder – Definition, Truth table, obtaining simplified expression for sum and carry ,Circuit Diagrams, obtaining full adder from half adders.

## **UNIT V: COMPUTER NETWORKS (07 MARKS)**

- **Networks:**  
Definition, Components (Nodes, Server, Network Interface Unit), Need for Networking.
- **Types of Networks:**  
LAN, MAN, WAN- Basic concepts.
- **Communication Channel:**  
Physical Channel: Twisted Pair Cable, Co-axial Cable, Optical Fibre Cable (Diagram, description, application).  
Wireless Channel: Microwave, Radio wave, Satellite Links.
- **Data Switching Techniques:**  
Circuit Switching , Message Switching and Packet Switching.
- **Data Communication Terminologies:**  
Baud, Baud rate, Bit rate, Bandwidth, Data Transfer rate, Bit rate (KBPS, MBPS, GBPS, TBPS)
- **Network Devices and their uses:**  
Modem, Hub, Repeaters, Bridge, Router , Gateway, Switch.
  
- **Network Topologies:**  
Definition, Types of Topologies (Bus, Tree, Star, Ring).
- **Client Server Model:**  
Concept of Client, Server ,Client Server Model, and Backbone Network.
- **Protocols:** Definition, File Transfer Protocol (FTP), Hyper Text Transfer Protocol (HTTP), Transmission Control Protocol/Internet Protocol(TCP/IP), Simple Mail Transfer Protocol(SMTP), Post Office Protocol(POP), Remote Login(Telnet).
- **Application of Networks:** Email, E-commerce, Chat Services, Video Conferencing, Usenet.
- **Internet Related Terminologies:**  
Internet, Requirements of Internet, Internetworking, Internet Service Providers , Internet Addressing, World Wide Web(WWW), Uniform Resource Locator(URL), Web Server, Webpage, Web Server, Website, Web Browser, Hyper Text Mark-up Language ( HTML), Dynamic Hyper Text Mark-up Language (DHTML), Extended Mark-up Language (XML), Search Engine, Downloading and Uploading files on/from the net, Hacking, Cracking, Cookies.

## **SUGGESTED ASSIGNMENTS (any one)**

1. Programs in C++ (Minimum 2), involving concepts related to Classes & Objects, Function Overloading, Inheritance, Constructors and Destructors.
2. Problem solving in Stack, Queues, Arrays and Linked list.

For Example-Evaluation of postfix expression  
Conversion from infix to postfix expression  
Problems on Stacks and Queues etc.

3. Programs on basic file operations (Read,Write,Update, Search, Append etc)
4. Problems based on SQL (Given a table ,writing queries using SQL commands)
5. Problems based on: K-MAPS  
Implementation of NAND –NOR & NOR-NAND logic  
Designing circuits using universal gates.
6. Presentations can be prepared on any topic related to computer networks.

**N.B. Viva can be conducted on any of the related topics**

The criteria for the evaluation of the assignments should be based on the following. (As applicable for the appropriate assignments).

1. Understanding of concepts
2. Knowledge w.r.t. the assignment given
3. Content (for presentations)
4. Logic (for programs)
5. Test Cases- Programs should be tested for different set of inputs.

## SHORT PROGRAMS LIST

1	Write a C++ program to <b>reverse</b> a given positive integer <b>M</b> , of arbitrary length (i.e. any number of digits) and output <b>M</b> as well as its reverse.
2	Write a C++ program to calculate <b>Sin(x)</b> , by summing the terms of the sine series given below. <b>N</b> is the number of terms to be summed and <b>x</b> is real number specifying an angle in radians. Also check the result of your calculation with that of the standard library function <b>sin()</b> . $\text{Sin}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} \dots \text{up to } N \text{ terms}$
3	Write a C++ program to generate the first <b>N</b> terms ( <b>N</b> ≥ 1) of the <b>Fibonacci series</b> 0, 1, 1, 2, 3, 5.....
4	Write a C++ program to <b>count the no of vowels</b> in a given line of text. Output the line of text and the number of vowels in it.
5	Define a class <b>NUMBER_LIST</b> having the following specifications: Private data member: num_list- an array of integers capable of storing maximum 10 numbers. Public member functions: read_list()- to read <b>N</b> ( <b>N</b> ≤ 10) numbers in the array num_list. calc_product()- to compute the product of the numbers in the array and display the numbers and their product.
6	Write a C++ program to create an array of maximum size 10 and storing integer values. Input <b>N</b> ( <b>N</b> ≤ 10) numbers into the array. Process the array to find and output the <b>largest and the smallest</b> numbers from the array as well as all the values in the array.
7	Write a C++ program to <b>transpose</b> a <b>m x n</b> matrix of integers, where ( <b>m</b> ≠ <b>n</b> ) and output the original input matrix and its transpose.
8	Define a class with two private data members <b>d1</b> and <b>d2</b> of type <b>double</b> and one private member function <b>smaller()</b> , that returns the smaller of the two values. Define two public member functions as follows: get_data(double, double)- to assign values to the data members. put_data() - to display the data values and the <b>smaller of the two values</b> by calling <b>smaller()</b> .
9	Define a class <b>BASE</b> having one private data member <b>num1</b> and one public data member <b>num2</b> both of type float. Define public member functions : <b>input_data()</b> - to read data value <b>num1</b> . <b>get_num1()</b> - to return the value of <b>num1</b> Extend class <b>BASE</b> to another class <b>DERIVED</b> using public derivation. Define for class <b>DERIVED</b> , a private data member <b>sum</b> which is to be calculated by adding <b>num1</b> and <b>num2</b> and a public member function : <b>get_data()</b> - to read <b>num2</b> and to call <b>input_data()</b> for reading value to and to compute sum. <b>show_data()</b> - to output <b>num1</b> , <b>num2</b> and <b>sum</b> Write a <b>main()</b> to create object of type <b>DERIVED</b> and input and output all data.

10	Write a C++ program to create two text files named: <b>COUNTRY</b> and <b>CAPITAL</b> to store names of n countries and their corresponding capitals. Then read the files to display the names of their countries and their capitals in a neat tabular form.
----	--

### LONG PROGRAM LIST

1	Write a menu driven program in C++ to create an array of N integers sorted in ascending order and to search for a given value in the array using <b>Binary Search</b> algorithm.
2	Write a C++ program to input two arrays A and B of integers , both sorted in ascending order . <b>Merge</b> A and B to obtain a third array C, such that C is also in ascending order. Output all the arrays A , B and C
3	Write a C++ program to input two matrices having integer elements – M1 of order (m x n)and M2 of order (n x p) and obtain the <b>product matrix</b> P. Output all the three matrices M1, M2 and P.
4	Write a menu driven program in C++ to create an array of N numbers and to sort the array in ascending order using <b>Bubble Sort</b> technique. Output the array before and after sorting.
5	Write a menu driven program in C++ to create an array of N numbers and to sort the array in ascending order using <b>Selection Sort</b> technique. Output the array before and after sorting.
6	Write a menu driven program in C++ to create an array of N numbers and to sort the array in ascending order using <b>Insertion Sort</b> technique. Output the array before and after sorting.
7	Write a menu driven program in C++ to push and pop a value in a linked stack storing float point numbers. Display the contents of the <b>stack</b> after each operation.
8	Write a menu driven program in C++ to add and remove a value in a linked queue storing float point numbers. Display the contents of the <b>queue</b> after each operation.
9	<p>Write a menu driven program in C++ to create and display a linked list having N nodes, where the data part consist of</p> <p>empcode- integer  empname- of maximum length 20 characters  basic_pay- float  allowance- float</p> <p>The output should be in a neat tabular form as shown below:</p> <pre> ***** Sr.No   Emp Code   Emp Name   Basic Pay   Allowance   Total   1      101      ..... ***** </pre>

10	<p>Define a class <b>STUDENT</b> having the following specifications:</p> <p>Private data members:</p> <ul style="list-style-type: none"><li><b>rollno</b>- integer</li><li><b>name</b> -of maximum length 35characters</li><li><b>marks</b>-an array of size 6 storing marks in 6 subjects</li></ul> <p>Public member functions:</p> <ul style="list-style-type: none"><li><b>input()</b>-to read the data values</li><li><b>show()</b>- to compute the overall percentage of each student and to display all data Values.</li></ul> <p>Write a C++ program to create and display a binary file having objects of type <b>STUDENT</b>.</p>
----	---

## DESIGN OF THE QUESTION PAPER (FIRST FORMATIVE EXAM 2014 ONWARDS)

Class : XII (General Stream)

Time : 1 Hrs.

Subject : Computer Science

Max. Marks : 20

The weightage or the distribution of marks over different dimensions of the question paper shall be as follows:

### 1. Weightage to learning Outcomes

Sr.No.	Learning Outcomes	Marks	Percentage of Marks
1	Knowledge	06	30%
2	Understanding	09	45%
3	Application	05	25%
4	Skill	--	--
Total		50	100%

### 2. Weightage to Content/Subject Units

Sr.No.	Units	Marks
1	Unit I :- Programming in C++ (only Interitance)	5
2	Unit II :- Data structures in C++ - Arrays	10
	- Linear linked lists	5
Total		20

### 3. Weightage to Forms of question

Sr.No	Forms of Question	Marks for each Question	Number of Questions	Total Marks
1	Long Answer Type(LA)	04	01	04
2	Short Anaswer Type(SA-I)	02	04	08
3	Short Answer Type(SA-II)	03	01	03
4	Very Short Answer Type(VSA)	01	05	05
Total			11	20

The expected time different type of question would be as follows:

Sr.No	Forms of Question	Approx. Time for each Question in mins (t)	Number of Questions	Approx. Time for each Question in mins (n x t)
1	Long Answer Type(LA)	12	01	12
2	Short Answer Type(SA)	07	04	28
3	Short Answer Type(SA-II)	10	01	10
4	Very Short Answer Type(VSA)	02	05	10
Total			11	60

As the total time is calculated on the basis of the number of questions required to be answered and the length of their anticipated answers, it would, therefore, be advisable for the candidates to budget their time properly by cutting out the superfluous words and be within the expected time limits.

#### 4. Scheme of Options

There will be no overall choice. However, there is an internal choice in 01 sub questions of 04 marks category.

#### 5. Weightage to difficulty level of questions:

Sr. No.	Estimated difficulty level of questions	Percentage
1	Easy	20.00%
2	Average	60.00%
3	Difficulty	20.00%

A question may vary in difficulty level from individual to individual . As such, the assessment in respect of each question will be made by the paper on the basis of general anticipation from the group as a whole taking the examination. This provision is only to make the paper balanced in its weightage, rather than to determine the pattern of making at any stage.

#### 6. Number of Main Questions :

There will be 02 main questions of 10 marks each

# NOTE: MCQ's should not be included in the question paper

Signature of the convener



## DESIGN OF THE QUESTION PAPER (SECOND) FORMATIVE EXAM 2014 ONWARDS)

Class : XII (General Stream)

Time : 1 Hrs.

Subject : Computer Science

Max. Marks : 20

The weightage or the distribution of marks over different dimensions of the question paper shall be as follows:

### 2. Weightage to learning Outcomes

Sr.No.	Learning Outcomes	Marks	Percentage of Marks
1	Knowledge	06	30%
2	Understanding	09	45%
3	Application	05	25%
4	Skill	--	--
Total		50	100%

### 2. Weightage to Content/Subject Units

Sr.No.	Units	Marks
1	Unit I :- Programming in C++ (only Interitance)	5
2	Unit II :- Data structures in C++ - Arrays	10
	- Linear linked lists	5
Total		20

### 3. Weightage to Forms of question

Sr.No	Forms of Question	Marks for each Question	Number of Questions	Total Marks
1	Long Answer Type(LA)	04	01	04
2	Short Anaswer Type(SA-I)	02	04	08
3	Short Answer Type(SA-II)	03	01	03
4	Very Short Answer Type(VSA)	01	05	05
Total			11	20

The expected time different type of question would be as follows:

Sr.No	Forms of Question	Approx. Time for each Question in mins (t)	Number of Questions	Approx. Time for each Question in mins (n x t)
1	Long Answer Type(LA)	12	01	12
2	Short Answer Type(SA)	07	04	28
3	Short Answer Type(SA-II)	10	01	10
4	Very Short Answer Type(VSA)	02	05	10
Total			11	60

As the total time is calculated on the basis of the number of questions required to be answered and the length of their anticipated answers, it would, therefore, be advisable for the candidates to budget their time properly by cutting out the superfluous words and be within the expected time limits.

#### 4. Scheme of Options

There will be no overall choice. However, there is an internal choice in 01 sub questions of 04 marks category.

#### 5. Weightage to difficulty level of questions:

Sr. No.	Estimated difficulty level of questions	Percentage
1	Easy	20.00%
2	Average	60.00%
3	Difficulty	20.00%

A question may vary in difficulty level from individual to individual . As such, the assessment in respect of each question will be made by the paper on the basis of general anticipation from the group as a whole taking the examination. This provision is only to make the paper balanced in its weightage, rather than to determine the pattern of making at any stage.

#### 6. Number of Main Questions :

There will be 02 main questions of 10 marks each

# NOTE: MCQ's should not be included in the question paper

Signature of the convener