Rao IIT Academy
Symbol of Excellence and Perfection
JEE | MEDICAL-UG | BOARDS | KVPY | NTSE | OLYMPIADS

XII HSC - BOARD - MARCH - 2017

Date: 20.03.2017

COMPUTER SCIENCE - I (D-9)
SOLUTIONS

1. (A)

(a) (iii) Context Switching

Topic:Operating System; Sub-topic:Context Switch_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) (iii) Fields

Topic:Data Structure; Sub-topic: Field_Target-2017_XII–HSC Board Exam __Computer Science-I

(c) (iii) Private

Topic:C++ Programming; Sub-topic: Access Specifier_Target-2017_XII–HSC Board Exam __Computer Science-I

(d) (ii)

Topic:HTML; Sub-topic:Tag_Target-2017_XII–HSC Board Exam __Computer Science-I

1. (B)

(a) (1) C++ allows the common function to be made friendly with more than one classes, there by allowing the function to have access to the private data of classes. Such a function need not be member of any of classes.

(2) Non-member function cannot have access to the private data of a class. However, there could be a situation where user would like two classes to share a particular function. At this situation friend function is used.

(3) The keyword "friend" declares the function to be friendly with that class. This function is defined as a normal C++ function. The function definition does not use class-name, keyword friend or scope resolution operator.

(4) A friend function has following characteristics:

(i) It is not in the scope of the class to which it has been declared as friend.

(ii) Since it is not in scope of the class, it cannot be called by using object of that class. It is called like a normal C++ function.

(iii) It can be declared either in public or the private part of a class without affecting its meaning.

(iv) Usually, it has the objects as arguments.

(v) It cannot access the member function directly and has to use and object name and dot operator with each member name.

Topic:C++ Programming; Sub-topic: Friend Function_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) The three special characteristics of a static data member in a class are as follows:

(1) It is initialized to zero when the first object of its class is created. No other initialization is permitted.

(2) Only one copy of that member is created for the entire class and is shared by all the objects of that class, no matter how many objects are created.

(3) It is visible only within the class, but its life time is the entire program.

Topic:C++ Programming; Sub-topic: Static Member_Target-2017_XII–HSC Board Exam __Computer Science-I

(c)

(1) Virus detection

(i) Normally virus detection program checks integrity of binary files. It maintains a checksum on each file. At regular frequency detection program calculates checksum and matches with original one. If there is mismatch then that program may be infected.

(ii) Some programs reside in the memory and continuously monitor memory and I/O operations against virus.

(2) Virus removal

There are some viruses whose bit pattern in the code can be predicted. The virus removal program scans the disk for the patterns of known viruses and on detection it removes them.

(3) Virus prevention

For prevention of virus the user can take the following precautions

(i) Always buy legal copies of software.

(ii) Take frequent backups of data

(iii) Run monitor programs frequently to detect virus.

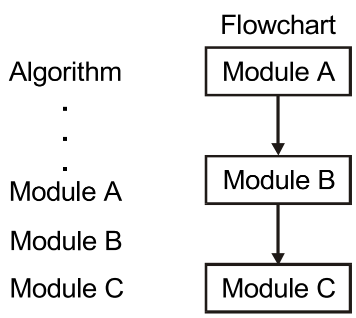
Topic:Operating system; Sub-topic:Virus_Target-2017_XII–HSC Board Exam __Computer Science-I

2. (A)

(a) (i) **Sequence Logic:**

(1) In the sequence logic modules are executed sequentially one after another.

(2) The sequence is represented by means of numbered steps. The flow pattern is as shown in figure.



(3) The above three modules are executed in sequence that is one after another.

(ii) **Selection Logic:**

(1) This logic employs a number of conditions and these conditions select the module among several alternative modules.

These are the types of conditional structures:

(2) **Single Alternative:**

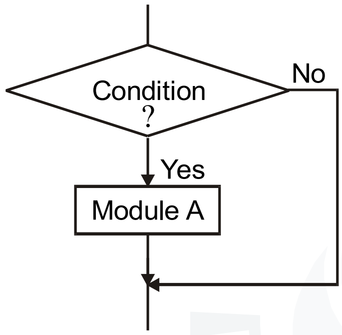
If condition holds true then Module A is executed, otherwise Module A is skipped and control transfers to the next step of algorithm.

Form of the structure:

If condition, then:

[Module A]

[End of IF Structure]



(2) Double Alternative

This structure has following form

IF condition, then:

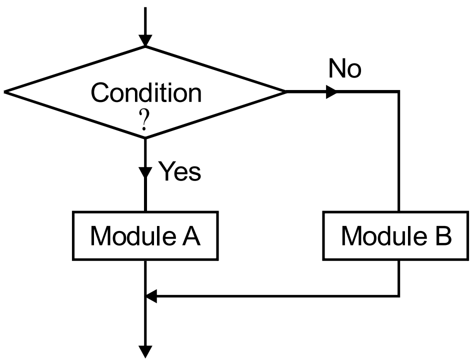
[Module A]

Else:

[Module B]

[End of IF Structure]

if condition is true Module A is executed and if condition is false Module B is executed.



(iii) **Interation Logic:**

The repeat for loop has the following form,

Form:

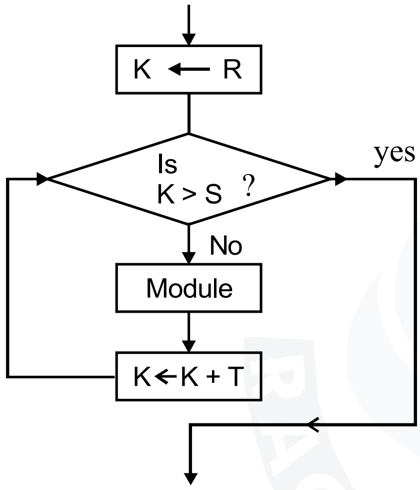
Repeat for K = R to S by T
[Module]
[End of loop]

where K = Index variable

R = Initial value of K

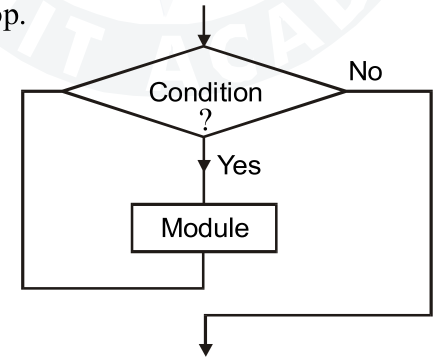
S = Final value of K

T = Step (increment/decrement)



The loop is executed for all values of K starting from R to S with step of T.

It uses a condition to control the loop.



Form:

Repeat while condition:
[Module]
[End of loop]

The loop continues till a condition is true. There must be initialization statement and there must be a statement that will change index variable in loop.

Topic:Data Structure; Sub-topic:Algorithm_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) **Algorithm:**

[Bubble Sort] BUBBLE [DATA, N]

Here DATA is an array with N elements. This algorithm sorts the elements in DATA.

(1) Repeat steps 2 and 3 for K = 1 to N-1

(2) Set PTR: = 1 [pass pointer PTR.]

(3) Repeat while PTR<= N–K. [Executes pass.]

(a) IF DATA [PTR] > DATA [PTR + 1], then:
Interchange DATA [PTR] and DATA [PTR + 1]
[End of IF structure]

(b) Set PTR: = PTR + 1
[End of inner loop.]
[End of step 1 outer loop.]

(4) Exit.

Example:

Consider the array as follows:

DATA[1]	56
DATA[2]	43
DATA[3]	05
DATA[4]	07
DATA[5]	09

Pass 1:

(a) Compare DATA [1] with DATA [2] since 56>43 So it is exchanged,

	56	43	5	7	9
New list is	43	56	5	7	9

(b) Compare DATA [2] with DATA [3] since 56 >5 so it is exchanged,

	43	56	5	7	9
New list is	43	5	56	7	9

(c) Compare DATA [3] with DATA [4] since 56>7 so it is exchanged,

	43	5	56	7	9
New list is	43	5	7	56	9

(d) Compare DATA [4] with DATA [5] since 56>9 so it is exchanged,

	43	5	7	56	9
New list is	43	5	7	9	56

At the end of first pass, the largest element 56 has moved to the last position.

Pass 2 : As K=2 , the comparisons will be 3.

New list is	43	5	7	9	56
(a)	43	5	7	9	56 since 43>5 exchange
	5	43	7	9	56
(b)	5	43	7	9	56 since 43>7 exchange
	5	7	43	9	56
(c)	5	7	43	9	56 since 43>9 exchange
	5	7	9	43	56

Pass 3:

5	7	9	43	56	No exchange
5	7	9	43	56	No exchange

New List:

5 7 9 43 56

Pass 4:

In this way after complete execution of this algorithm, the array gets sorted in Ascending order as follows:

DATA[1]	5
DATA[2]	7
DATA[3]	9
DATA[4]	43
DATA[5]	56

Topic:Data Structure; Sub-topic:Algorithm_Target-2017_XII–HSC Board Exam __Computer Science-I

(c) **(1)** To keep track of all memory locations free or allocated and if allocated, to which process and how much.

(2) To decide memory allocation policy i.e., which process should get how much memory when and where.

(3) To use various techniques and algorithms to allocate or deallocate memory locations.

Normally, this is achieved with the help of some special hardware.

The following are the memory management systems :

(A) Contiguous, Real Memory Management system :

(a) Single contiguous

(b) Fixed partitioned

(c) Variable partitioned

(B) Non- contiguous, Real Memory Management System :

(a) Paging

(b) Segmentation

(c) Combined

(C) Non - contiguous, Virtual Memory Management System :

(a) Virtual memory

Topic:Operating System; Sub-topic:Memory management_Target-2017_XII–HSC Board Exam __Computer Science-I

(B)

(a) (1) To define an additional task to an operator, it specify what it means in relation to the class to which the operator is applied. This is done with the help of a special function, called operator function, which describes the task.

(2) In short, a function which defines additional task to an operator or which gives a special meaning to an operator is called the operator function.

(3) The general form of operator function is,

```
return-type class-name : : operator op(argument list)
{
    function body//task defined
}
```

Where return type is the of value returned by the specified operation and op is the operator being overloaded.

The op is preceded by the keyword operator. **Operator op is** the function name.

(4) Operator functions must be either member functions (non-static) or friend functions.

(5) The basic difference between operator function as a friend function and as member function is that a friend function will have only one argument for unary operators and only one for binary operators. This is because the object used to invoke the member function is passed implicitly and therefore is available for the member function. This is not the case with friend function. Arguments may be passed either by value or by reference.

Topic:C++ Programming; Sub-topic:Operator function_Target-2017_XII–HSC Board Exam __Computer Science-I

(b)

(i) External Priority: - The user specifies the priority externally at the time of initiating the process. If the user does not specify any priority, operating system assumes a default priority for that.

If the job is too urgent, system manager permits the process at a higher priority.

(ii) Purchased Priority: In this priority higher priority processes are charged at a higher rate. The operating system keeps the track of the time used by each process and charges it accordingly.

(iii) Internal priority: It is decided by scheduling algorithms. Their calculations are based on the current state of the process. The following scheduling algorithms are used for that.

(a) **Shortest Job First Algorithm:** The job whose expected time for completion is less is executed first.

(b) **Expected remaining time to complete:** It is same as SJF at the beginning but as the process progresses the time will change. At regular intervals operating system calculates the expected remaining time for completion and accordingly priority is determined.

(c) **Time Slice**: Each process is given a certain time for execution. The predetermined time given is called as time slice of the process.

Topic:Operating system; Sub-topic:Priority_Target-2017_XII–HSC Board Exam __Computer Science-I

3. (A)

(a)

	Linear Search		Binary Search
1.	Linear search performs on unsorted list of elements as well as sorted list.	1.	For binary search, the elements in array are stored in alphabetically or numerically in sorted manner.
2.	Compare the desired element with all elements in an array until the match is found.	2.	Compare the value of midpoint with desired value. If the value is greater than midpoint value, the first half is checked, otherwise second half checked until search is successful or interval empty.
3.	Insertion of an element in an array can be performed very efficiently when array is not ordered.	3.	An insertion of a new element requires that many elements be physically moved to preserved order.
4.	For large size of array, time required for this search is very larger.	4.	For large size of array, comparatively time required is less.
5.	Time complexity is as follows : Worst case : N comparison Best case : 1 comparison	5.	Time complexity as follows: Worst case : $\log_2$ N comparison Best case : 1 comparison

Topic:Data Structure; Sub-topic:Search Algorithm_Target-2017_XII–HSC Board Exam __Computer Science-I

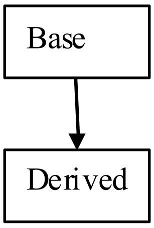
(b)

(1) The mechanism of deriving a new class from an old one is called as inheritance. The old class is referred as base class and new class is referred as derived class. C++ strongly supports the concept of reusability. This is basically done by creating new classes, reusing the properties of the existing ones. Functions and variables of a class that has been tested can be used by object of another class. This is known as inheritance.The reuse of a class that has already been tested, debugged and used many times, can save the efforts of developing and testing the same again.

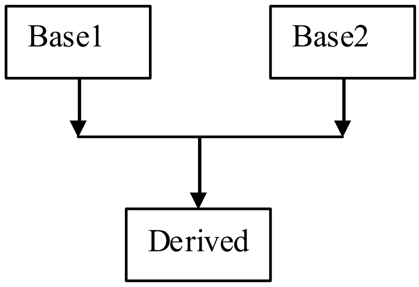
(2) One of the most useful features of classes is inheritance. It is possible to declare a class that inherits the properties of another class or classes. This means that, with good class design, you can build applications, which are based on proven re-usable code. i.e. main purpose behind inheritance is code-reusability.

(3) Different forms of Inheritance:

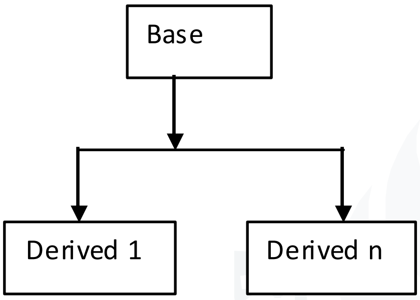
1. Single inheritance



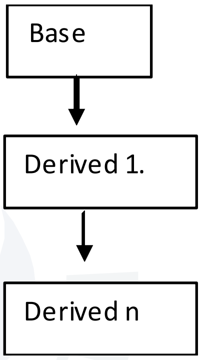
2. Multiple Inheritance



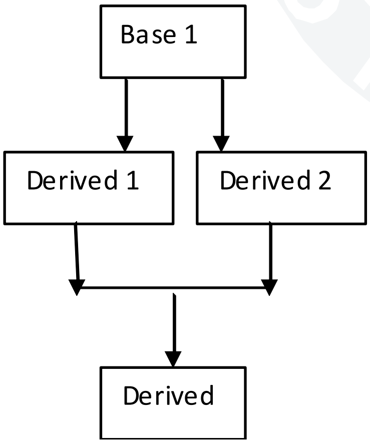
3. Hierarchical Inheritance:



4. Multilevel Inheritance



5. Hybrid Inheritance



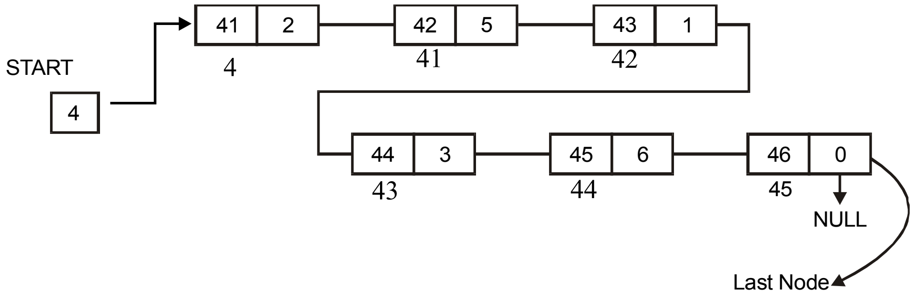
Topic:C++ Programming; Sub-topic: Inheritance_Target-2017_XII–HSC Board Exam __Computer Science-I

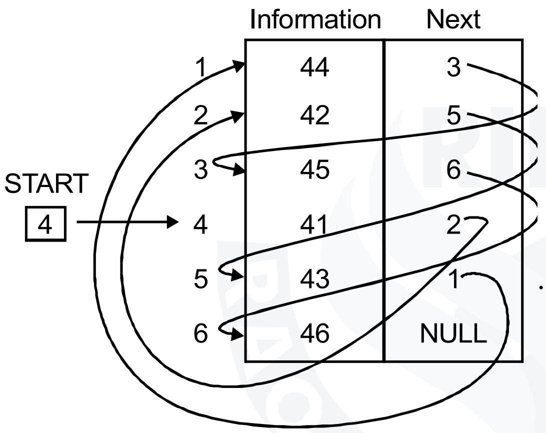
(c)

(1) Linked lists can be represented in memory by using two arrays respectively known as INFO and LINK, such that INFO[K] and LINK[K] contains information of element and next node address respectively.

(2) The list also requires a variable 'Name' or 'Start', which contains address of first node. Pointer field of last node denoted by NULL which indicates the end of list. e.g., Consider a linked list given below :

(3) The linked list can be represented in memory as -



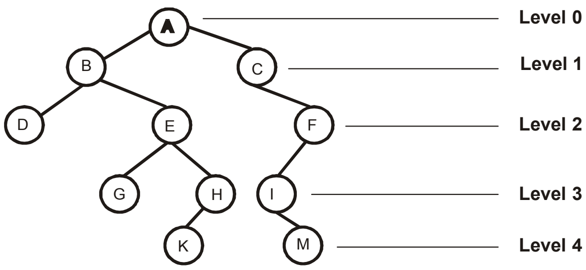


Above figure shows linked list. It indicates that the node of a list need not occupy adjacent elements in the array INFO and LINK.

Topic:Data Structure; Sub-topic:Link List_Target-2017_XII–HSC Board Exam __Computer Science-I

3. (B)

(a)



(i) **Root :** A node which has no parent. The node containing 'A' is the root of the tree.

(ii) **Leaf :** The node which has no child (or children) .D, G, K, M are leaves.

(iii) **Sibling :** Two nodes have the same parent. The nodes containing D and E are both children of the node containing B. These nodes are siblings.

(iv) **Depth :** Depth of a tree is defined as maximum level of any nodes in the tree. If root is level 0 then depth or height of tree is equal to 1 + largest level number.

For eg: Depth of above tree is 5.

Topic:Data Structure; Sub-topic:Tree_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) (1) **Computer Virus:** It is not a complete program by itself. It can not act independently. It is a program written with a clear intention of infecting other programs.

(2) **Infection Methods:**

(a) **Append:** In this method viral code appends itself to the unaffected program.

(b) **Replace:** In this method viral code replaces original executable program completely or partially.

(c) **Insert:** In this method the viral code is inserted in the body of an executable code to carry out some funny actions.

(d) **Delete:** In this case viral code deletes some codes from executable program.

(e) **Redirect:** In this case the normal control flow of a program is changed to execute some other code.

Topic:Operating System; Sub-topic:Virus_Target-2017_XII–HSC Board Exam __Computer Science-I

4. (A)

(a) (1) Every HTML document has the same general structure, and it consists of few tags that define the page as a whole.

(2) The primary part of an HTML document are denoted by <HTML>, <HEAD> and <BODY> tags. Each of these tags are known as **Document Structure Tags**.

(3) HTML file always starts with <HTML> tag. Similarly ended with </HTML> tag. It declares text within web page viewed in a web browser.

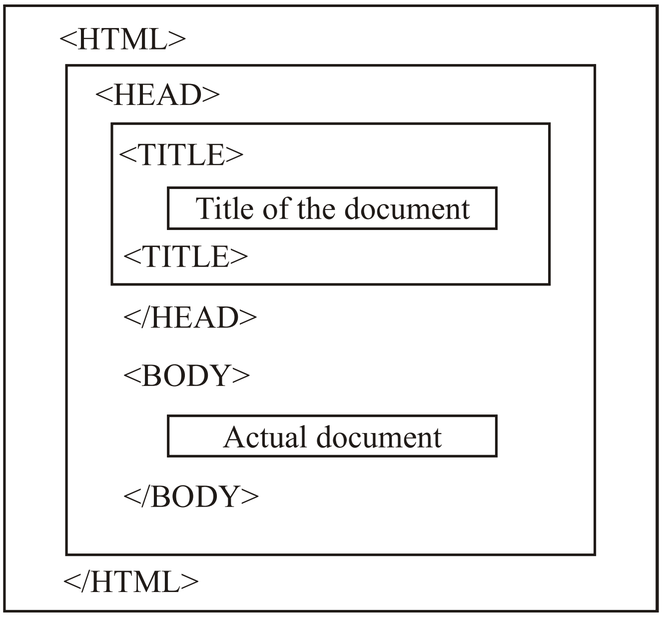
(4) HTML document can be divided into two sections:

(a) **The head:** It is like an introduction to the page. It generally consists of title of the page. To define head, add <HEAD> tag at beginning and </HEAD> tag at end of heading.

(b) **The body** : In this user enters the text images and other tags that will actually appear on the web page.

To define the body, place <BODY> tag at beginning and </BODY> tag at the end after the head section.

These tags define the basic structure of every web page.



Topic:HTML; Sub-topic: Basic Tags_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) (1) The use of same function name to create functions that perform a variety of different tasks is called as function overloading.

(2) Overloading refers to the use of same thing for different purposes. Function overloading or function polymorphism, is an example of compile time polymorphism.

(3) Using the concept of function overloading, create a family of functions with one function name but with different argument lists.

(4) The function would perform different operation, depending on argument list in function call.

(5) The correct function to be invoked is determined by checking the number and the type of the arguments and not on the function type.

(6) e.g.
```
#include<iostream.h>
int area (int s);         //prototype declaration
int area (int l, int b);  //for overloading area( )
void main ( )
{
cout<<area (10); //function call
cout <<area (5, 10);
}
int area (int s)      //function definition
{
return (s*s);
}
int area (int 1, int b)
{
    return (1*b);
}
```
In above example the function area ( ) is overloaded. The first function is used to calculate area of square. It has one integer parameter.

The second function is used to calculate area of rectangle. It has two integer parameters.

(7) When a function is called, the compiler first matches the prototype having same number and types of arguments and then calls appropriate function for execution. A best match must be unique.

Topic:C++ Programming; Sub-topic:Function Overloading_Target-2017_XII–HSC Board Exam __Computer Science-I

(c) (1) It is multitasking, multi-user and multithreading operating system.

(2) It also supports virtual memory management system to allow multiprogramming.

(3) Symmetric multiprocessing allows to it to schedule various tasks on any CPU in a multiprocessor system.

(4) It uses New Technology File System which implements fault tolerance, security and support for large files.

(5) It is 32 bit operating system.

Topic:Operating System; Sub-topic:Windows NT_Target-2017_XII–HSC Board Exam __Computer Science-I

(B)

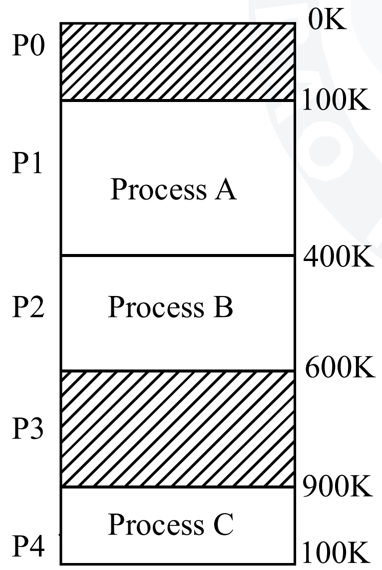
(a) (1) It is the simplest method for memory allocation. It divides the memory into fixed size sections. These sections are called as partitions. Each partition may contain exactly one process. When the partition is free, a process is selected from the input queue and loaded into free partitions. When the process terminates, the partition becomes available for another process.

(2) The partitions are fixed at the time of system generation. It is the process of tailoring the operating system to specific requirements. After declaring partitions the operating system creates a partition description table.

(3) **The Memory Map for processes is as follows:**

Process: P0-(0-200K)
Process: P1-(200-700K)
Process: P2-(700-800K)
Process: P3-(800-1100K)
Process: P4-(1100-1500K)

(4) **Partition Description Table:**



Fixed partition

Partition ID	Partition		
	Starting Address	Size	Status
0	0	100K	FREE
1	100	300K	ALLC
2	400	200K	ALLC
3	600	300K	FREE
4	900	100K	ALLC

PDT

(5) Each process block contains process ID. If the block is allocated, it is shown allocated. Otherwise it is free and it is shown as free.

Topic:Operating system; Sub-topic:Partition_Target-2017_XII–HSC Board Exam __Computer Science-I

(b) **Use of Scope resolution Operator (::)**

The scope resolution operator (**::**) is used to access the data of global version of variable. C++ is a block-structured language. Different blocks can be used in a program. The same variable name can be used in different blocks with different values. They are stored at different memory locations.

Example:

```
#include<iostream.h>
void main( )
{
 int a=10;          // local to main
cout<<"a="<<a;    // a=10
{
int a=25; // local to inner block
cout<<a; // a=25
cout<<::a;            // a=10   outer block variable is accessed
}
}
```

Use of memory management operator : new and delete

The **new** operator is used to create objects of any type. The syntax is as follows:

Pointer_variable = new data type;

Example : p = new int; // p is pointer of type int .

The **delete** operator is used to destroy the variable from the memory space so that the same space can be used for another use. The syntax of delete is as follows:

delete pointer variable;

Example: delete p; // delete pointer variable p from memory.

Topic:C++ Programming; Sub-topic: Operators_Target-2017_XII–HSC Board Exam __Computer Science-I

Q.5

(a) //Program to find out largest number from the given array

```
#include<iostream.h>
void main ( )
{
    int num [10], max;
    cout>>"Enter the number",
    for (int i = 0; i < 10; i++)
    cin<<num [i];
    max = num [0];
    for (int j = 1; j<10;j++)
{
    if(max<num [j])
```

```
        max = num [j];
    }
        cout<<"The largest number in the array is" <<max;
    }
```

Topic:C++ Programming; _Target-2017_XII–HSC Board Exam __Computer Science-I

(b) //Program to find factorial of a number

```
#include<iostream.h>
#include<<conio.h>
void main ( )
{
    int fact, number;
    clrscr ( ) ;
    fact = 1;
    cout << "Enter the number" <<endl;
    cin >> number;
    for (i = 1; i < = number; i++)
{
    fact = fact*i;
}
    cout<< "The factorial of a inputted number is" <<fact;
}
```

Topic:C++ Programming; _Target-2017_XII–HSC Board Exam __Computer Science-I

(c) HTML code is an follows:

```
<HTML>
<HEAD>
<TITLE> COMPUTER PAPER ANALYSIS </TITLE>
</HEAD>
<BODY>
<TABLE BORDER = 2 WIDTH = 100%
                    CELLSPACING = 15>
<TR>
    <TD     WIDTH = 25% ROWSPAN = 2 ALIGN = CENTER>
                COMPUTER<BR> SCIENCE </TD>
    <TD     WIDTH = 25% ALIGN = CENTER > PAPER - I </TD>
    <TD     WIDTH = 25% ALIGN = CENTER > PAPER - II </TD>
```

```
        <TD    WIDTH = 25% ALIGN = CENTER > TOTAL </TD>
    </TR>
    <TR>
        <TD    WIDTH = 25% ALIGN = CENTER > 100  </TD>
        <TD    WIDTH = 25% ALIGN = CENTER > 100  </TD>
        <TD    WIDTH = 25% ALIGN = CENTER > 200  </TD>
    </TR>
    </TABLE>
    </BODY>
    </HTML>
```

Topic:HTML; _Target-2017_XII–HSC Board Exam __Computer Science-I

OR

(a)
```
#include<iostream.h>
    class gcd
{
    int a, b;
    public :
    void print ( );
};
    void gcd :: print (void)
{
cin >> a >> b;
while (a ! = b)
{
    if (a > b)
        a – = b;
        if (b>a)
        b – = a;
}
        cout << a;
void main ( )
{
gcd obj;
obj.print ( );
}
```

Topic:C++ Programming; _Target-2017_XII–HSC Board Exam __Computer Science-I

(b) //Program to generate fibonacci series

```
#include < iostream.h>
void main ( )
{
int f0, f1, f, n;
f0 = 0;
f1 = 1;
clrscr ( );
cout<<"Fibbonacci series\n";
cout<<"n" <<f0<<"\n"<<f1;
for (n=1; n<=20; n++)
{
 f=f0 + f1;
 cout<<"\n"<<f;
 f0 = f1;
        f1 = f;
}
}
```

Topic:C++ Programming; _Target-2017_XII–HSC Board Exam __Computer Science-I

(c)

Introduction ⟵ (Title of the page)

Computer Science ⟵ (Text size h1 in bold default font)

Paper - I ⟵ (Text size is default, regular, default font is used, underlined word)

Paper - II ⟵ (Text size is default, regular, default font is used, underlined word)

Topic:HTML; _Target-2017_XII–HSC Board Exam __Computer Science-I