Rao IIT Academy
Symbol of Excellence and Perfection

XII-HSC BOARD
COMPUTER SCIENCE - I (D-9)
SOLUTIONS

Q.1 (A)

a. (ii) Seek time **(1)**

b. (iv) Linked list **(1)**

c. (iv) Bottom up **(1)**

d. (i) NAME **(1)**

Q. 1 (B)

a. Virtual memory: **(1)**

1. Virtual memory is an attempt, which makes the execution of the processes possible, which may completely not in the main memory. Some part of the process may be on disk.
2. The MM techniques are simple to implement but the major drawback is that if the physical memory is limited then number of processes it can hold at any time i.e. degree of multiprogramming reduces. For this concept of virtual memory is introduced.

Elements of virtual memory. **(2)**

1. Locality of reference
2. Page fault
3. Working set
4. Page replacement policy
5. Dirty page
6. Demand paging

Page fault : When a page, which is not in main memory is referenced, then a page fault arises. At this time the O.S loads that page in main memory.

Working set : The set of pages in physical memory which are actively referred to any moment is called as working set. Working set helps to decide page replacement policy.

b. Data Structure: **(1)**

1. The logical of mathematical model of a particular organization of data is called data structure.
2. The data structure should be simple and it should show the relationship between data elements.

Array : An array is a data structure which allows a collective name to be given to a group of elements which all have the same type. An individual element of an array is identified by its own unique index (or subscript). An array can be thought of as a collection of numbered boxes each containing one data item. **(1)**

Pointer Array : An array is called pointer array, if each element of that array is a pointer. **(1)**

c. (i) **<IMG>** : is image tag.Its purpose is to include graphic images in the body of the web page. **(1)**
ex: <IMG SRC = " Mickey Mouse.jpg " ALT="MICKEY" >

iii. **Data Encapsulation :** **(1)**

The wrapping up of data and its associated functions together under a single unit is called as encapsulation.

Q.2 (B)

a. **System calls :** System call provided the interface between and the operating system. These calls are generayly available as assembly language instructions. **(1)**

List of system calls for memory management : **(1)**

1. To allocate a chunk of memory to a process.
2. To free chunk of memory from a process.

List of system calls for information management : **(1)**

1. Create a file
2. Create a directory

List of system calls for process management : **(1)**

1. Read a process
2. Block a process.

b. **Virtual Function:** **(1)**

When one class inherits its properties to other class and both derived class as well as base class has same function name, then it becomes difficult to access the right function name. In this case, the base version of the function is made virtual and we can access the right function for using its address.

Rules for Virtual Functions : **(3)**

When virtual functions are created for implementing late binding, we should observe some basic rules that satisfy the compiler requirements:

1. The virtual functions must be members of some class.
2. They cannot be static members.
3. They are accessed by using object Pointers.
4. A virtual function can be a friend of another class.
5. A virtual function in a base class must be defined, even though it may not be used.
6. The prototypes of the base class version of a virtual function and all the derived class versions must be identical. If two functions with the same name have different prototypes, C++ considers the as overloaded functions, and the virtual function mechanism is ignored.

Q.3 (A)

a. **Multi user operating system :** In multi user operating system support multiple processes at the same time. Time-sharing systems are multi-user systems. Most batch processing systems for mainframe computers may also be considered "multi-user", to avoid leaving the CPU idle while it waits for I/O operations to complete.

Multi-user software is software that allows access by multiple users of a computer. **$(1\frac{1}{2})$**

Time sharing operating system : It uses CPU scheduling and multiprogramming to provide each user with a small portion of time - shared computer. Each user has a separate program in memory. When program executes, it executes for only short time before it either finishes or needs to perform I/O. **$(1\frac{1}{2})$**

```
Step 2:  Repeat steps 3 and 4
         while BEG = END AND DATA (MID) ≠ ITEM
Setp 3:  If ITEM < DATA [MID], then :
         set  END = MID - 1
         Else :
         Set BEG = MID + 1
         [End of If structure]
Step4 :  Set MID = INT (BEG + END)/2)
         [End of step 2 loop]
Step5 :  If DATA [MID] = ITEM, then :
         set LOC = MID
         Else :
            LOC = NULL
         [End of If structure]
Step6 :  Exit
```

e.g. Given DATA be the following sorted 13 element array : **(1)**

```
11    22    30    33    40    44    55
60    66    77    80    88    99
```

Suppose ITEM = 40

```
Step1:   Intially BEG = 1 and END = 13
         Hence MID = INT [(1 + 13)/2] = 7
         and so DATA [MID] = DATA [7] = 55
Step2 :  Since 40 < 55, END has its value changed by
         END = MID - 1 = 7 - 1 = 6
         Hence MID = INT [(1 + 6)/2] = 3
         and so DATA [MID] = DATA [3] = 30
Step3 :  Since 40 > 30, BEG has its value changed by
         BEG = MID + 1 = 3 + 1 = 4
         Hence MID =  INT [(4 + 6)2/] = 5
         and so DATA [MID] = DATA [5] = 40
```

∴ found ITEM in location LOC = MID = 5

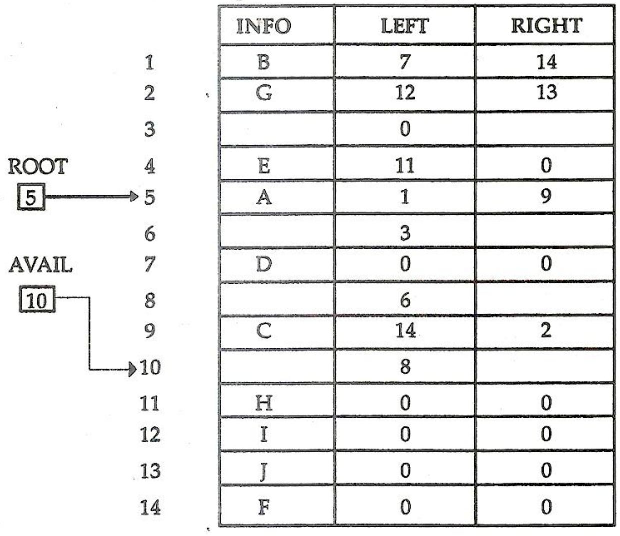
b. **Constructor :** **(1)**

1. A constructor is a special member function of a class. Its task is to initialise the objects of its class.
2. It is special because its name is same as that of the class to which it belongs.
3. The constructor is invoked whenever an object of its associated class is created.

e.g. A constructor is declared and defined as follows **(1/2)**

```
//class with constructor
class integer
{
        private:
        int m, n;
        public:
        integer (void); // constructor declared
};
integer : : integer (void) //constructor defined
{
```

The above tree can be represented in memory as,



	INFO	LEFT	RIGHT
1	B	7	14
2	G	12	13
3		0	
4	E	11	0
5	A	1	9
6		3	
7	D	0	0
8		6	
9	C	14	2
10		8	
11	H	0	0
12	I	0	0
13	J	0	0
14	F	0	0

(2)

The roots stores address of first node of tree T.

AVAIL stores address of first null node. To insert another node to tree T, it is inserted memoy location pointed by AVAIL.

c. The mechanism of deriving a new class from an old one is called as inheritance. **(1)**

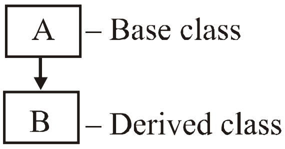
The old class is referred as base class and new class is referred as derived class.

There are five types of inheritances in C++:

1. **Single inheritance:** **(1)**

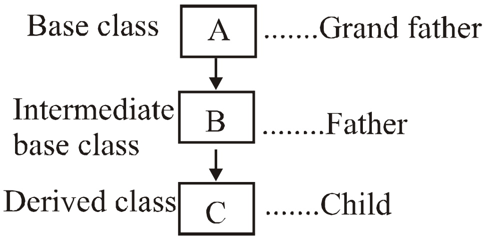
A derived class with only one base class is called as single inheritance.

It has the form :



2. **Multilevel inheritance:** **(1)**

The mechanism of deriving one class



Q.4 (B)

a. The main features are listed below:

Easier to use : **(1)**

1. Windows 98 is a single user multitasking operating system.
2. Navigating around the computer is easier in window 98.
3. A file can be opened by a single click.
4. Windows 98 allow us to use multiple monitors with single computer.
5. New hardware can be easily installed and used without restarting comptuer.
6. With Window 98 we can use digital cameras and other digital imaging devices.

Q.5 Answer any two of the following :

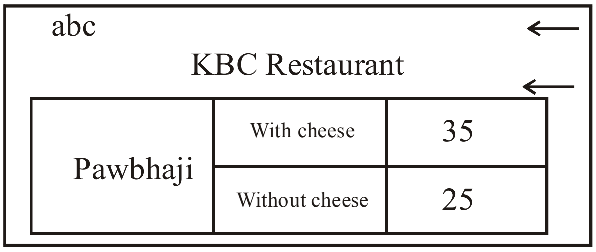
a. **(5)**

```
void great (int, int);
void main ( )
{
        int a, b;
        cout <<"Enter two no=";
        cin>> a>>b ;
                great {a, b};
        getch ( );
}
void great (int p,  int q)
{         if  (p  > q)
          {       cout << p << "is big" ;}
        else
          {      cout << q << " is big" ;}
}
```

b. **(5)**

```
void main ( )
{       int a, b, c, big;
        cout <<" Enter three no =";
         cin >> a >> b >> c;
         big  = a;
         if  (b > big)
          {  big = b;}
         if   (c > big)
          {  big = c; }
        cout <<"\n largest number  = " <<big;
          getch ( );
}
```

c. Output of HTML program **(5)**



abc

KBC Restaurant

Pawbhaji	With cheese	35
	Without cheese	25

[title]

[Center align H1 text]

OR

a. **(5)**

```
void main ( )
{       char a[80], b[80];
         int  i;
        i = 0;
                cout <<" Enter a string=";
                 cin >> a;
```

```
                    <LI> Commerce
                        <UL Type = “Circle”>
                            <LI> ENGLISH  </LI>
                            <LI> ACCOUNTS </LI>
                        </UL> </LI>
                </OL>
            </BODY>
    </HTML>
```