

- I. **Q.1** (a) Differentiate between call by value and call by reference, giving suitable examples. 2
1. Differentiate between primary memory and secondary memory.. Give examples of each type of memory. 2
- A)**The memory inside the CPU is primary memory (main memory) and the memory outside it is known as secondary(auxiliary) memory.
- Primary Memory:** RAM (Random Access Memory) and ROM (Read only Memory) comes under primary memory. RAM is volatile memory and ROM is non volatile memory. All the data and programs must be stored in RAM for execution. But the content of RAM is not permanent.  
Eg: RAM, ROM.
- Secondary Memory:** Since primary memory has a limited storage capacity and is not permanent, secondary storage devices are used to store large amount of data permanently. There are various types of secondary devices available these days.  
Eg: Hard disks, Floppy disks ----- Magnetic Media  
CD ROMs, DVDs ----- Optical Media
2. Expand the following: 2
- i)CPU ii) ROM iii)MICR iv)CD-R
- A)** (i) CPU - Central Processing Unit  
(ii)ROM - Read Only Memory  
(iii)MICR - Magnetic Ink Character Reader  
(iv)CD -R - Compact Dist Recordable
3. Explain briefly the functions of ALU. 2
- A)** The Arithmetic and Logic Unit performs all the four arithmetical (+,-,\*,/) and some logical (<,<=,>,>=,< >) operations. When two numbers are required to be added, these numbers are sent from memory to ALU where addition takes place and the result is put back in the memory. In the same way other arithmetic operations are performed.
- For logical operations also, the numbers to be compared are sent from memory to ALU where the comparison takes place and the result is returned to the memory. The result of a logical operation is either TRUE or FALSE. These operations provide the capability of decision-making to the computer.
- II.1.** What is the difference between semantics error and syntax error? Give an example of each. 3
- A) 1.Syntax Errors:** Syntax errors occur when rules of a programming language are misused.  
**Syntax** refers to formal rules governing the construction of valid statements in a language.  
**Eg:** int a,b //Did not keep ; (semicolon) at the end of statement.
- 2.Semantics Error:** Semantic errors occur when statements are not meaningful.  
**Semantics** refers to the set of rules which give the meaning of a statement.  
**Eg:** X \* Y = Z;  
(Siva plays Guitar is Syntactically and Semantically correct but  
Guitar plays Siva is Syntactically correct but Semantically incorrect).
2. Write an algorithm to find  $x^y$ . 2
- A)** Step1: Start  
Step2: Declare x, y as integers  
Step3: Output "Enter the value of x and y ie base and exponent"

Step 4: Read x,y

Step 5: Output "The result of the expression =", pow(x,y)

Step 6: Stop

3. What is data abstraction? Explain the concept with the help of an example. 2

**A) Abstraction:** Abstraction refers to the act of representing essential features without including the background details or explanations.

Eg: If you talk about student details, complete details are his rollno, admissionno, name, m1, m2, m3, total, avg, grade, fathename, address, pin, etc.....

But abstraction about his marks are...

Rollno, admissionno, name, m1, m2, m3, total, avg, grade.

4. Expand UPS. 1

**A) Uninterrupted Power Supply**

5. How many bytes are in a terabyte? 1

**A) 1024 X 1024 X 1024 X 1024 Bytes**

(or)  $2^{10} \times 2^{10} \times 2^{10} \times 2^{10}$  Bytes (or)  $2^{40}$  Bytes

6. Write the names of two popular system softwares.

1

**A) Operating Systems: DOS, Windows XP, Unix, Solaris.**

Language Processors: Assembler, Compiler, Interpreter.

II. 1. What do you mean by a lexical unit? Give an example. 1

**A) Token:** The smallest individual unit in a program is known as a token (**Lexical Unit**).

There are 5 types of tokens.

(i) Keywords (ii) Identifiers (iii) Literals (iv) Punctuators (v) Operators

2. In each of the following cases show how the comment can be placed in a program:

i) Add a comment **sum of three numbers** to the statement:

sum=a+b+c;

**A) sum = a+b+c; //sum of three numbers**

ii) Add the comment **End of Function** with the '}'.

2

**A) } //End of Function**

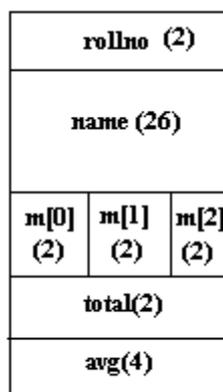
3. Differentiate between unions and structures with help of an example. 2

**A) Structure:** A structure is a collection of elements (variables) of different data types.

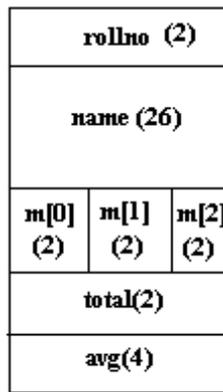
The keyword **struct** is used to construct a structure.

**Syntax:** struct structtag  
{ datatype variable;  
datatype variable;  
-----  
-----  
} structure variables;

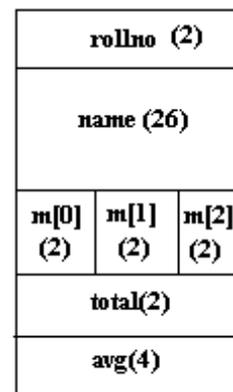
**Eg:** struct student  
{ int rollno;  
char name[26];  
int m[3];  
int total;  
float avg;  
} s1, s2;  
struct s3;



s1



s 2



s 3

**Referencing Structure variables:**

Structurevariable.elementname;

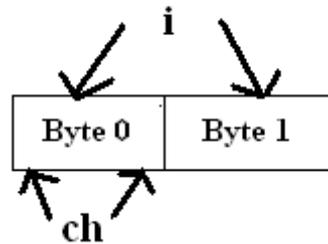
Eg: s1.rollno;

```
s2.name;
s3.marks[2];
```

**Union:** A union is a memory location that is shared by two or more different variables (generally shared by two or more different times).

Eg: union share

```
{ int i;
  char ch;
};
union share us;
us.i = 20;
cout<<us.ch;
```



At any point, you can refer to the data stored in a integer or a character.

**us** as either an

**Difference between a union and a structure:**

A **union** is a memory location that is shared by two or more different variables.

A **structure** have separate memory location for every variable.

In the above example for structure variable S1, 40 Bytes will be reserved and For union variable us, 2 bytes will be reserved.

**4. Differentiate between object oriented programming and procedural oriented programming with the help of examples of each.**

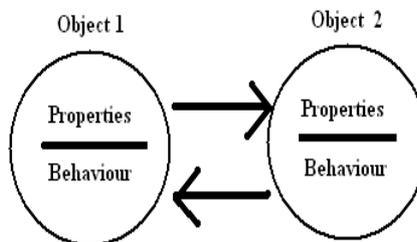
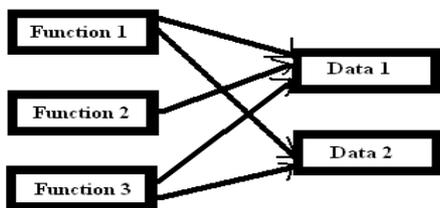
2

**A. Procedure Oriented Programming:**

A program in a procedural language is a list of instructions where each statement tells the computer to do something. The focus is on the processing. There is no much security for the data.

**Object Oriented Programming:**

The object oriented approach views a problem in terms of objects involved rather than procedure for doing it. In object oriented programming, object represents an entity that can store data and has its interface through functions.



All data is openly available to all functions in the program

Data and functions enclosed within objects. New objects communicate with one another.

OOP is so closer to the real life. And also it consists many features like

1.Data Abstraction 2.Data Encapsulation 3.Modularity 4.Inheritance 5.Polymorphism 6. Data hiding.

**5. Distinguish between if and switch statement.**

2

**A)**The if-else and switch both are selection statements and they both let you select an alternative out of given many alternatives by testing an expression.

But there are some differences in their operations.

(i) The switch statement differs from the if statement in that switch can only test for equality whereas if can evaluate a relational or logical expression. Ie multiple conditions.

(ii) The switch statement selects its branches by testing the value of same variable whereas the if else construction lets you use a series of expressions that may involve unrelated variables and complex expressions.

(iii) The if-else can handle ranges whereas switch cannot. Each switch case label

Must be a single value.

(iv) The if-else statement can handle floating point tests also where as switch cannot handle floating point tests.

(v) The switch case label value must be a constant. So, if two or more variables are to be compared, use if-else.

6. Differentiate between call by value and call by reference with help of an example. 4

A)(i) In call by value, actual arguments will be copied into the formal parameters.

In call by reference, formal parameters are references to the actual arguments.

(ii) In call by value, if any modification is occurred to the formal parameter, that change will not reflect back to the actual argument.

In call by reference, if any modification is occurred to the formal parameter (reference to the actual argument), the actual argument value will be changed.

(iii) We should go for call by value when we don't want to modify the original value.

We should go for call by value when we want to modify the original value.

(iv) Example:

```
void Change(int a, int &b)
{
    a= 2*a;
    b=20;
}
```

Here a is called by "call by value" method and b is called by "call by reference"

So as the value of a is changed, actual argument for a will not be changed,

as the value of b is changed, actual argument for b will be changed

7. Write the names of the header files of the following functions. 2

i) getch() ii) isalpha() iii) strcpy() iv) sqrt()

A) (i) getch() - conio.h (ii) isalpha() - ctype.h

(iii) strcpy() - string.h (iv) sqrt() - math.h

### III.

1. Write a program to print the diagonal (left & right) elements of an N×N matrix. 4

A) // Program to print the left and right diagonal element of an NXN matrix

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
#include<iomanip.h>
```

```
void main()
```

```
{ int A[10][10];
```

```
int i,j,N;
```

```
clrscr();
```

```
cout<<"\nHow many rows and columns required for matrix: ";
```

```
cin>>N;
```

```
cout<<"\nEnter "<<N*N<<" elements: ";
```

```
for(i=0;i<N;i++)
```

```
{ cout<<"Enter the elements into Row "<<i+1<<": ";
```

```
for(j=0;j<N;j++)
```

```
cin>>A[i][j];
```

```
}
```

```
clrscr();
```

```
cout<<"\nThe entered elements in the matrix are: \n";
```

```

for(i=0;i<N;i++)
{ for(j=0;j<N;j++)
    cout<<A[i][j]<<"\t";
  cout<<endl;
}
cout<<"\n\n\nThe elements which are belongs to only diagonals...\n";
for(i=0;i<N;i++)
{ for(j=0;j<N;j++)
    if((i==j) || ((i+j)==(N-1)))
        cout<<setw(6)<<A[i][j];
    else
        cout<<" ";
  cout<<endl;
}
getch();
}

```

2. Write a program to find the factorial of a number recursive function.

3

```

A) #include<iostream.h>
    #include<conio.h>
    long f=1;
    long factorial(int n)
    { if (n==0)
        return f;
      else
        f=n*factorial(n-1);
    }
    void main()
    { clrscr();
      long num;
      cout<<"\nEnter the number to which you want to find factorial: ";
      cin>>num;
      cout<<"\nThe factorial of the number = "<<factorial(num);
      getch();
    }

```

3. Write a program to find the total number of characters, lines and words in a paragraph of text.

4

```

A) #include<iostream.h>
    #include<conio.h>
    #include<stdio.h>
    void main()
    { char str[300];
      int i,charcount=0,words=1,lines=1;
      clrscr();
      cout<<"\nEnter the Paragraph ie message: \n";
      gets(str);
      for(i=0;str[i]!='\0';i++)

```

```

{ charcount++;
  if(str[i]!=' ')
    words++;
  if (charcount%80==0)
    lines++;
}
cout<<"\nNumber of Characters in the entered message: "<<charcount;
cout<<"\nNumber of Words in the entered message: "<<words;
cout<<"\nNumber of Lines in the entered message: "<<lines;
getch();
}

```

4. Write a program to sort an array on N numbers in ascending order. Avoid duplication of elements.

3

```

A) #include<iostream.h>
#include<conio.h>
void main()
{ clrscr();
  int A[20],N,i,j,temp;
  cout<<"\nEnter the number of elements:";
  cin>>N;
  for(i=0;i<N;i++)
    cin>>A[i];
  //Bubble sort technique
  for(i=0;i<N;++i)
    for(j=0;j<(N-1)-i ;j++)
      if(A[j]>A[j+1])
        { Temp=A[j];
          A[j]=A[j+1];
          A[j+1]=Temp;
        }
  cout<<"The Elements in the array after sorting.... ";
  for(i=0;i<N;i++)
    cout<<A[i]<<'t';
}

```

5. Write a program to find the roots of a quadratic equation.

2

```

A) #include<iostream.h>
#include<conio.h>
#include<math.h>
void main()
{ clrscr();
  double d1,d2,b,a,c,d;
  cout<<"\nEnter the value of b,a and c: ";
  cin>>b>>a>>c;
  d=(b*b-sqrt(4*a*c));
}

```

```

    if(d==0)
        cout<<"\nRoots are equal or distinct";
    else if(d>=0)
        cout<<"\nRoots are Real";
    else
        cout<<"\nRoots are complex..ie Imaginary";
    d1=(-b+d)/(2*a);
    d2=(b+d)/(2*a);
    cout<<"\nD1: "<<d1;
    cout<<"\nD2: "<<d2;
    getch();
}

```

6. Write a program to convert the temperature reading in Fahrenheit to Celsius. 2

```

A) #include<iostream.h>
#include<conio.h>
void main()
{ double c,f;
  clrscr();
  cout<<"\nEnter the temperature in Fahrenheit: ";
  cin>>f;
  c=(5/9.0)*f-32;
  cout<<"\nThe equivalent temperature in Celcius: "<<c;
  getch();
}

```

#### IV.

1. What is the difference between '=' and '=='? 1

A) = is assignment operator. This operator stores the right side value into left side variable.

== is relational operator. It will check whether right side and left side operands are equal or not and returns a true(1) when both are equal, false(0) otherwise.

2. What will be the result of the following two expressions if i=10 initially? 2

i) cout<<i<<i++<<++i;

A) 121111

ii) cout<<i<<i+1<<++i;

A) 111211

3. What will be the output of the following program? 2

```

int main()
{
  int i=0,x=0;
  for(i=1;i<10;i*=2)
  {
    x++;
    cout<<x;
  }
  cout<<"\n"<<x;
}

```

A) 1234

4

4. Rewrite the following program after removing the syntactical error(s) if any.

Underline each correction.

2

```
#include<iostream.h>
int main
{ struct movie
{ char movie_name[20];
char movie_type;
int ticket_cost=100;
}movie;
gets(movie_name);
gets(movie_type);
}
```

A) #include<iostream.h>

#include<stdio.h>

```
void main()
{ struct movie
{ char movie_name[20];
char movie_type;
int ticket_cost;
}Movie;
Movie.ticket_cost=100;
gets(Movie.movie_name);
cin>>Movie.movie_type;
}
```

5. Convert the following equations to C++ statements.

2

2

i)  $s = 1 + 1/x + 1/x^2 + 1/x^3$

A)  $s = 1 + 1/x + 1/(x*x) + 1/(x*x*x);$

ii)  $V = 4/3\pi r^3$

A)  $V = 4/(3*3.1415*r*r*r);$

6. Explain any two string handling functions with syntax and examples.

2

A) **String functions:**

Function	Description	Example
1. strcat - string concatenation. char *strcat (char *str1, const char *str2)	This function concatenates a copy of str2 to str1 and terminates str1 with a null. str1 should be large enough to hold both its original contents and those of str2	char *str1="One"; char *str2="Two"; strcat(str2,str1); cout<<str2; (It will print "TwoOne").
2. strcmp - string compare int strcmp (const char *str1, const char *str2)	This function alphabetically compares two strings and returns -ve value if str1 is less than str2, 0 if str1 is equal to str2, and >0 (+ve value) if str1 is greater than str2.	char *str1="Ace"; char *str2="Bag"; strcmp(str2,st1) returns +ve value (>0 value) but strcmp (str1,str2), return -ve value.

3. strcpy - string copy char *strcpy(char *str1, const char *str2)	Copies the contents of str2 into str1	char *str1="Ace"; char *str2="Bag"; strcpy(str1,str2); cout<<str1; (It will print "Big");
4 strlen - string length int strlen(char *str)	Returns the length of the null terminated string pointed to by str. The null is not counted.	strlen("Raju") would give 4.

V. 1.  $(AC.20)_{16} = (?)_2 = (?)_8$

2

A)

$(AC.20)_{16}$

Binary equivalent of A = 1010

Binary equivalent of C = 1100

Binary equivalent of 2 = 0010

Binary equivalent of 0 = 0000

So,  $(AC.20)_{16} = (10101100.00100000)_2$

**10 | 101 | 100 | . 001 | 000 | 00**

$(10101100.00100000)_2 = \mathbf{2\ 5\ 4\ 1\ 0\ 0} = (254.100)_8$

2.  $(4A56)_{16} = (?)_{10} = (?)_8$

2

A)  $(4A56)_{16} = 4 \times 16^3 + 10 \times 16^2 + 5 \times 16^1 + 6 \times 16^0$   
 $= 4 \times 4096 + 10 \times 256 + 5 \times 16 + 6 \times 1$   
 $= 16384 + 2560 + 80 + 6$   
 $= (19030)_{10}$

$(4A56)_{16} = (0100\ 1010\ 0101\ 0110)_2$

**0 | 100 | 101 | 001 | 010 | 110**

**= 4 5 1 2 6 = (45126)\_8**

3. Which of the following are hardware and software?

2

3

(i) Capacitor (ii) Internet Explorer (iii) Hard disk (iv) UNIX

A) (i) Capacitor - Hardware  
(ii) Internet Explorer - Software  
(iii) Hard Disk - Hardware  
(iv) UNIX - Software

4. Find the 1's and 2's complement of 128.

2

A) Binary Equivalent of 128 = 10000000

1's Complement of 128 = 01111111

+1

2's Complement of 128 = 10000000

5. Define the following:

2

i) DRAM ii) Serial Port

- A) (i) **DRAM:** Dynamic Random Access Memory. DRAM consists of a transistor and a capacitor that's capable of storing an electric charge. Depending on the switching action of the transistor, the capacitor either contains no charge(0) or does hold a charge(1). DRAM is volatile ie contents are lost in the event of power failure. Regenerator circuits are there to refresh the memory.
- (ii) **Serial Port (COM or RS232C Ports):** The serial port transfers data serially a bit at a time. Serial ports come in the form of 9 pin or 25 pin male connector.