[This question paper contains 7 printed pages.]

2516 *Your Roll No.* .....

B.Sc. (G) / II **A**

COMPUTER SCIENCE – Paper III

(Object Oriented Programming)

Time : 3 Hours *Maximum Marks : 38*

(Write your Roll No. on the top immediately on receipt of this question paper.)

Attempt **All** *Questions.*
Answer **all** *parts of a question together.*
Give explanation of the output wherever required.

1. Write the output(s) of the following :

(i)
```
Class Foo
{       static int i = 0;
        static int j = 0;
      public static void main(String args[ ])
    {   int i = 2;
        int K = 3;
      {  int j = 3;
         system.out.println("i+j is" +i+j);
         }
     K = i + j;
     system.out.println("K is" +K);
     system.out.println("j is" +j);
     }
}
```

(2)

P.T.O.

(ii) Class Test

```
{ Public static void main(String args[ ])
{ Count myCount = new Count( );
  int times = 0;
for(int i = 0; i < 100; i++)
  increment (myCount, times);
system.out.println("count is" + myCount.count);
system.out.println("times is" + times);
  }
public static void increment (Count C, int times)
{   C.count ++;
    times ++;
   }
  }
  Class Count
{   public int Count;
    Count (int C)
   {  count = C;}
    Count( )
    { count = 1;} }
```

(2)

(iii)
```
Class convert{
   public static void main(String args[ ])
{  byte b; int i = 257; double d = 232.14;
b = (byte)i;
system.out.println("b = " +b);
 i = (int)d;
 b = (byte)d;
system.out.println("i = " +i);
 system.out.println("b = " +b);
 }
}
```
(2)

(iv)
```
Class ConstructorChain{
   public static void main(String args[ ])
{  Child C = new Child( );}}
Class Child extends Parent {
Child( ) {system.out.println("Child( )constructor");}}
Class Parent extends Grandparent {
  Parent( ) {
       this(25);
   system.out.println("Parent( ) constructor");}
  Parent(int X) {
   system.out.println("Parent("+x+")constructor");}}
Class Grandparent{
  Grandparent( ) {
   system.out.println("Grandparent( )constructor");}}
```
(2)

(v)
```
Class Mix {
    int i;
  }
Class A extends Mix
{ int i;
  A (int a, int b)
  { super.i = a;
    i = b;
  }
  void show( )
  {
    System.out.println("here" + super.i);
    System.out.println("now" + i);
  }
}
Class Mixit {
  public static void main(String s[ ])
  {
      A Subob = newA(1, 2);
      Subob.show( );
  }
}
```
(2)

2. Find errors (if any) giving reasons :–

(i)
```
Class Demo1{
    public static void main(String args[ ])
    {    int x = 0;
         int n = 5;
         int y = 1;
         y = x + y;
       system.out.println("x=" +x); }
    system.out.println("y=" +y);
system.out.println("x=" = ((x+y)++));
}
```
(2)

(ii)
```
Private interface In{
      Void Method1( );
      int var1 = 2;
      int var2 = 5; }
Class Test implements In{
  public void method( ){
  system.out.println("Test version of In"); }
Class TestMain{
  public static void main(String args[ ]){
  Test t = new Test( );.
  System.out.println(var1);
t.var1 = t.var2;
  System.out.println("Value of var1 is :" + t.var1);
} }
```
(2)

P.T.O.

(iii)
```
Class demo{
    static void D1( )
{   try
    {   throw new NullPointerException("thrown");}
    Catch(NullPointerException e)
    { system.out.println("caught");
      throw e;}}
    public static void main(String args[ ])
    { D1( ); }}
```
(2)

(iv)
```
public Class Arg{
    String[ ] MyArg;
  public static void main(String arg v[ ]){
    MyArg = arg v;}
    public void amethod( ){
  system.out.println(arg v[1]); }}
```
(2)

3. (a) What is an applet ? Write an applet to print message "My first applet" at location 30, 50 as X, Y coordinate. Explain each statement used in the applet. (3)

(b) Explain Dynamic Dispatch method with example. (2)

(c) Write a java program to copy the contents of a file into another after removing the blank spaces from the file. (3)

4. (a) What are abstract classes ? How are they different from simple classes. What are the advantages of using them ? (3)

(b) Differentiate between :

(i) Finalize and finally

(ii) Throw and Throws (6)

(c) Define a class person having name as a data member. Inherit two more classes – Student and Employee from person. To the class student, add data members course, marks and year, and to the class employee, add data member department and salary. Write display( ) method in all the 3 classes to display the relevant detail of the corresponding classes. Provide the necessary method to show dynamic method dispatch concept. (3)